공학박사 학위논문

Open and Closed Contours Tracking Based on Shape Priors and Training

모양 정보 학습을 이용한 물체의 개방 혹은 폐쇄된 윤곽선 추적 방법

2016 년 8 월

서울대학교 대학원

전기컴퓨터공학부

허 선

ABSTRACT

This dissertation presents a new open and closed contours tracking algorithm using shape prior and its training based on a Bayesian framework, where the contour is a part (open contour) or the whole (closed contour) of the object's boundary. The shape of an object is a very important feature for many vision tasks such as object recognition and tracking. Specifically, the tracking performance can be increased if the target is determined and the tracker utilizes its shape information. The proposed method provides a new state space model for the representation of contours, which can reflect the shape information to the contour and handle rigid and non-rigid motions of contours independently. This model enables us to focus on the non-rigid motion during the tracking, and the model works for challenging rigid motion scenarios. In addition, for the robust tracking of contours, a measurement function that considers the contrast on object boundaries, target appearance, and temporal coherence is proposed. The proposed method is tested for various cases of contours such as open contour, closed contour and multi-contours. The state space model and measurement functions are modified a little bit in consideration of each contour model.

First, an open contour is modeled and tracked by the proposed method, which has received little attention during several decades compared with the closed contour

or bounding box shape tracking. The proposed state space model can represent an open contour that is moved by the dynamic model where rigid and non-rigid motions are absolutely separated. The measurement is designed with contrast, local track and appearance terms that indicate the proper position of the target and make the tracking more robust. The proposed method is applied to two examples of open contours targets (human omega shape and a cheetah profile), and experimental results show that the proposed method achieves superior performance to the conventional contour tracking methods. The proposed method is also compared with recent bounding box tracking methods for the object tracking purposes, and the comparison shows that the proposed method works robustly to fast motions and yields more accurate estimate of object's location than the conventional bounding box tracking methods.

Second, the proposed method is tested for the closed contour tracking which is usually carried out by segmentation algorithms or level set methods. A closed contour is described by the proposed model and deformed by the dynamic model. Measurement function is the same to the case of open contour tracking except the local track term, which is calculated with partial object appearances that are denoted by some local patches and their relative positions. As an application example, automobiles in blackbox video sequences are tracked by the proposed method. Experimental results show that the proposed method accomplishes higher performance than conventional tracking methods where some of them presents the target as a bounding box and others extract the object boundary using segmentation methods. Moreover, the document capture and tracking algorithm is also proposed, which is suitable for applying the proposed method because the shape of document is well known (a quadrilateral) and its boundary can be estimated by the proposed method.

This system is based on building quadrilaterals as document proposals using line segment detector and tests all proposals to find the best one with measurement terms. The proposed algorithm makes good marks at *2015 ICDAR comeptition*.

Finally, multi-contours tracking algorithm is devised based on the contour tracking method. It is assumed that targets belong to the same category and their appearances, colors and shapes are similar to each other. Thus, the proposed method trains only one shape model to track multi-contours. The state space vector is amended such that all contours can be represented by one state vector. In order to consider interactions between targets, the interaction term is attached to the existing dynamic model. As an example, human legs are tracked by the proposed method which may help to recognize the gaits. Experimental results show that conventional algorithms have troubles in tracking and distinguishing between the two legs, whereas all targets are well estimated accurately by the proposed method.

Key words: contour tracking, shape prior, open contour, multi-contours

Student number: 2010-20915

Contents

List of Figures

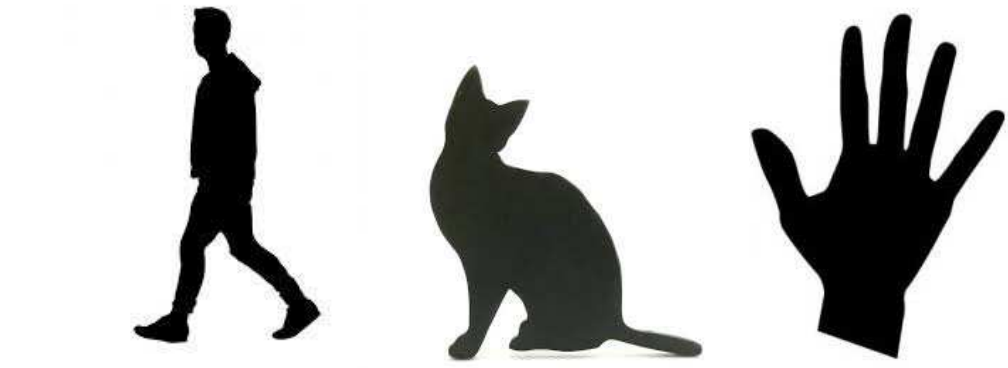
List of Tables

Chapter 1

Introduction

With the widespread of digital cameras and smart phones, a large number of videos are generated every day, and the analysis and effective use of contents in the videos have become important issues. One of the fundamental tools for video contents analysis may be the automatic object tracking, and the tracking methods can be classified into several categories according to their object representation methods (e.g., points, bounding boxes, geometric shapes like circles and ellipses, and contours), where most recent methods use the bounding box representation. Although the bounding box is simple and intuitive for the object representation, it has a weak point that only location information can be notified, without the object boundary which may be crucial and important information for some applications.

On the contrary, contour representations can allow us to obtain boundary information of the object as well as the object's position. Thus, a contour tracking algorithm can be a powerful tool for the tracking and also for some other applications. For example, the contour tracking algorithm can be applied to all the applications where the box tracking works such as surveillance system, traffic monitoring, sports



(a) (b) (c)

Figure 1.1: Silhouettes can notify many information of the object such as class and behavior. (a) a walking person, (b) a standing cat and (c) an open hand.

video analysis, pedestrian and vehicle tracking for the automatic vehicle system. In addition, the contour tracking works for the cases where the box tracking cannot be directly applied such as video editing, region of interest video coding, human computer interaction, and preprocessing for other vision algorithms. Therefore, even though there exists a fewer contour tracking algorithms compared with the popular bounding box tracking methods in recent days [1–12], many algorithms for contour tracking have been studied in several decades [13–19]. However, they are still far from satisfaction when the object's shapes and appearances are deformed non-rigidly due to their movements or view point changes, and also when the images have low quality owing to noise, blur, or background clutters and occlusions, or there is an illumination change over the sequences.

In order to track the object successfully, this dissertation focuses on the shape of the object because the shape is a very important characteristic of the object. As

shown Fig. 1.1, the class and the behavior of objects can be revealed only by the silhouette images. According to this perspective, a new contour tracking method based on shape priors is proposed and the shape prior is described in a new state space model. First, target dependent open contour tracking method is formulated where non-rigid motions are separated from rigid motions. Second, the proposed method models the target as a close contour and conduct the tracking based on its shape prior. Also, the proposed method estimates the closed contour without shape training for certain targets whose shapes are already well known. Finally, the proposed method is applied to multi-contours tracking where targets belong to the same class. Although there exists similar objects close to each other, the proposed method recognizes them accurately. In order to succeed the tracking for all contour targets, measurement terms are designed in consideration of the contrast of object's boundary, target appearance and temporal coherency. These terms are modified a little bit in each contour case.

In this dissertation, the proposed method adopts the recursive Bayesian estimation framework [15]. Specifically, Fig. 1.2 shows the overall flowchart of the algorithm, where a contour in the t-th frame is denoted as $\mathbf{r}^t$, which is generated by control points and basis functions as will be explained later in more detail. The control points are described by a state vector $\boldsymbol{\omega}^t$ that includes the rigid motion state $\mathbf{X}^t$ and nonrigid motion state $\mathbf{q}^t$. Here, rigid and nonrigid motions are defined on 2D space, so that the rigid motion indicates translation, in-plane rotation and scaling. The rotation in 3D space may be considered nonrigid motions because the 3D model of the target object is not generally available. The state vector at each frame

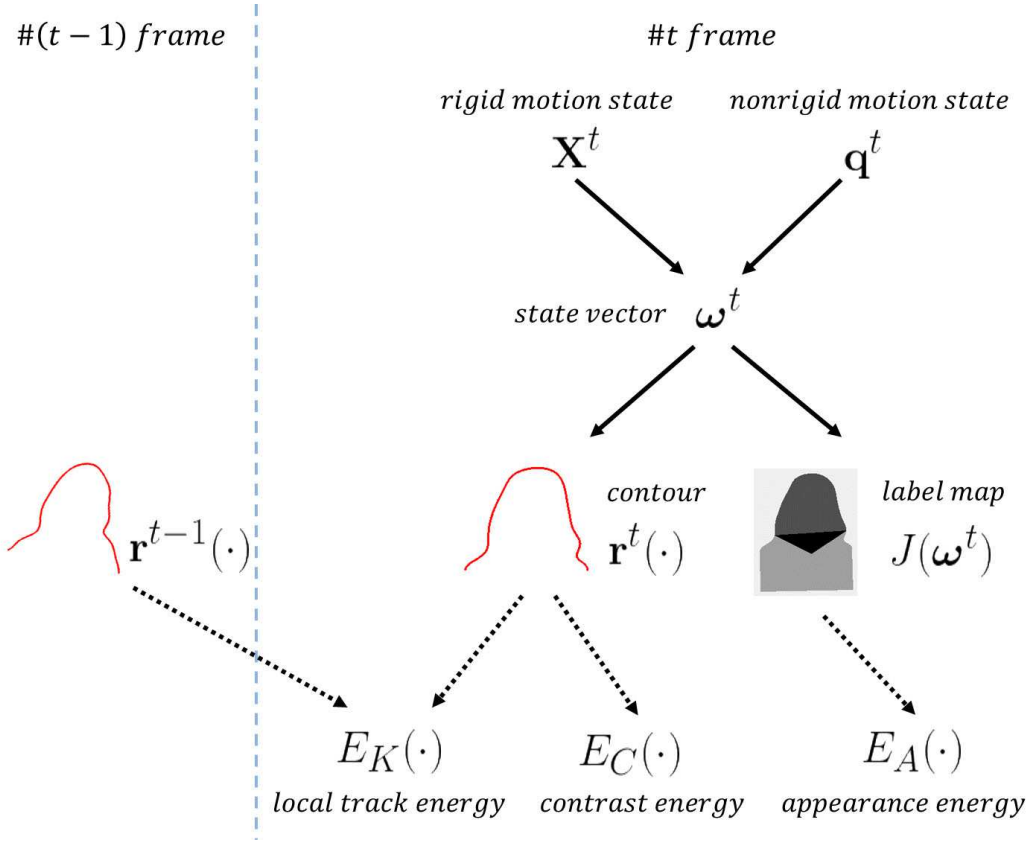


Figure 1.2: Notations in this dissertation. The state vector $\boldsymbol{\omega}^t$ represents the rigid motion state $\mathbf{X}^t$ and nonrigid motion state $\mathbf{q}^t$. From the state, the contour $\mathbf{r}^t(\cdot)$ is derived and the measurement function consisting of three complementary terms is evaluated for the tracking.

is updated as:

$$p(\boldsymbol{\omega}^t|I^{1:t}) \propto p(I^t|\boldsymbol{\omega}^t) \int p(\boldsymbol{\omega}^t|\boldsymbol{\omega}^{t-1}) p(\boldsymbol{\omega}^{t-1}|I^{1:t-1}) d\boldsymbol{\omega}^{t-1} \tag{1.1}$$

where I^t means the t-th frame and $I^{1:t} = \{I^1, \cdots, I^t\}$. In other words, the state vector $\boldsymbol{\omega}^t = \{\mathbf{X}^t, \mathbf{q}^t\}$ is evolved according to the change of rigid motion state $\mathbf{X}^t$ and nonrigid motion state $\mathbf{q}^t$, which is described by a stochastic dynamic model:

$$p(\boldsymbol{\omega}^t|\boldsymbol{\omega}^{t-1}) = p(\mathbf{X}^t|\mathbf{X}^{t-1})\ p(\mathbf{q}^t|\mathbf{q}^{t-1}). \tag{1.2}$$

The proposed method is based on this independence assumption (between the rigid and nonrigid motion), and this provides important benefits. First, a geometrically well-defined model (e.g., the model in [8]) for the rigid motion dynamics can be adopted, i.e., $p(\mathbf{X}^t|\mathbf{X}^{t-1})$, which allows us to handle challenging rigid motions without the training of such motions (e.g., 360° in-plane rotation that will be shown in each experimental section of chapters). Note that, in the conventional method [15], the state model cannot handle (rigid) motions that were not contained in training samples. Second, the learning of nonrigid motion dynamics $p(\mathbf{q}^t|\mathbf{q}^{t-1})$ in the training can be focused, whereas the conventional method needs to collect samples for all the possible rigid/nonrigid motions and their combinations. Actually, some of 3D model based tracking algorithms employed this decoupling idea [20, 21]. However, to the best of our knowledge, this is the first approach to decouple rigid and nonrigid motions in the 2D object tracking, and moreover, we exploit this decoupling in the state space model. In addition to the new state space model, a measurement function $p(I^t|\boldsymbol{\omega}^t)$ is also developed, to consider the appearance and temporal coherence of boundaries in the tracking.

The main contributions of this dissertation are followed :

- A new contour tracking algorithm is proposed for both open and closed contours, which is based on a new state model and a measurement function.
- The state model deals with rigid and nonrigid motions independently and its training method is also developed.
- The measurement function considers several complementary terms (contrast, appearance, and temporal coherence) for robust tracking.
- The interaction term is proposed for multi-objects and targets that belong in the same category are distinguished well during multi-contours tracking.
- The overall tracking system is efficient and it can track the contour in quasi-real time.

1.1 Open contour tracking based on a nonrigid shape training

The contours can be closed or open, where the closed contour is more often studied than the open one because the closed contour differentiates an object from the background. That is, there have been only a few studies on the open-contour tracking problem [15, 18, 19], and their applications are also very limited (e.g., microtubules and tongue tracking [18, 19]). However, the open contour tracking may be more appropriate than the bounding box or closed contour tracking for some applications such as tracking a human omega shape (head-shoulder), tracking the (partial) profiles of animals, or tracking the objects by parts. Also, open contour tracking can exhibit superior performance to the bounding box approach in tracking silhouettes that are distinctive features of objects as can be seen in Chapter 3.

In this chapter, the proposed method is applied to two kinds of targets (human omega shape and a cheetah profile) which are demonstrated by open contours. The state vector $\boldsymbol{\omega}^t$ consists of the rigid and non-rigid motion state: $\boldsymbol{\omega}^t = \{\mathbf{X}^t, \mathbf{q}^t\}$ where $\mathbf{X}^t$ and $\mathbf{q}^t$ are independent as shown in (1.2). The measurements are calculated by three energy terms: contrast term E_C, local track term E_K and appearance term E_A that definitions of theses terms are explained in Section 3.3.

1.2 Target-based closed contour tracking

In order to track the object whose shape is deformed non-rigidly, a closed contour is a good model to represent the object boundary and one of the successful approaches is active contour model based tracking methods [22, 23], which sets a contour over the object's boundary and tracks it over time rather than tracking the features directly. There are two different ways of representing object contour: one is the explicit method that parameterizes the contour, for example by the B-spline model as in [15], and the other is the implicit method like level set [13,24–27]. Most implicit representation methods have merits in describing the objects' deformation, merging and separation, and rigid transformation with only level set function, however, there exists a limitation to consider appearances and shapes of the object and the operation speed is usually slow. In the explicit representation methods, contour modeling is a difficult work, but if a contour is modeled once, then it is easy to handle contours.

In this chapter, a closed contour is represented by the proposed state model as the same framework to the open contour case in Chapter 3. The state vector consists of the rigid and non-rigid motion state: $\boldsymbol{\omega}^t = \{\mathbf{X}^t, \mathbf{q}^t\}$ where the elements are independent each other. The measurement also consists of three energy terms:

E_C, E_K and E_A where the local track term E_K is different from the case of open contour tracking. Partial object appearances are defined as a set of local patches and this set P^t is employed to measure the E_K.

Moreover, the proposed method can be simplified when the target shape is well known and also tested in that case. A document is chosen to the target and a new document capture system is proposed. Since we already know the document shape (a quadrilateral), the training process for shape priors is unnecessary. Automatic document capture is an interesting topic as a lot of camera captured document images are produced these days owing to the development of digital devices and it can be useful as the preprocessing step before other document vision algorithms such as text lines or pictures detector and optical character reader (OCR). The document is captured automatically and efficiently by the proposed method, as it uses well known shape information with tracking-by-detection manner.

1.3 Multi-contours tracking for objects that belong in the same category

In the last chapter of this dissertation, multi-contours are simultaneously described with a single state vector $\boldsymbol{\omega}^t$ where a part of the state vector for each contour $\boldsymbol{\omega}_i^t$ is concatenated to build the state vector $\boldsymbol{\omega}^t$. In addition to existing measurement function, the interaction term between contours $p(\boldsymbol{\omega}_i^t, \boldsymbol{\omega}_j^t)$ is added to consider relative locations among contours. Since all targets belong to the same category, they should have similar shapes and sizes. Also, in order to prevent several contours from converging one local position, targets prefer to move apart from one another. Furthermore, contours comply with the law of inertia that they choose their loca-

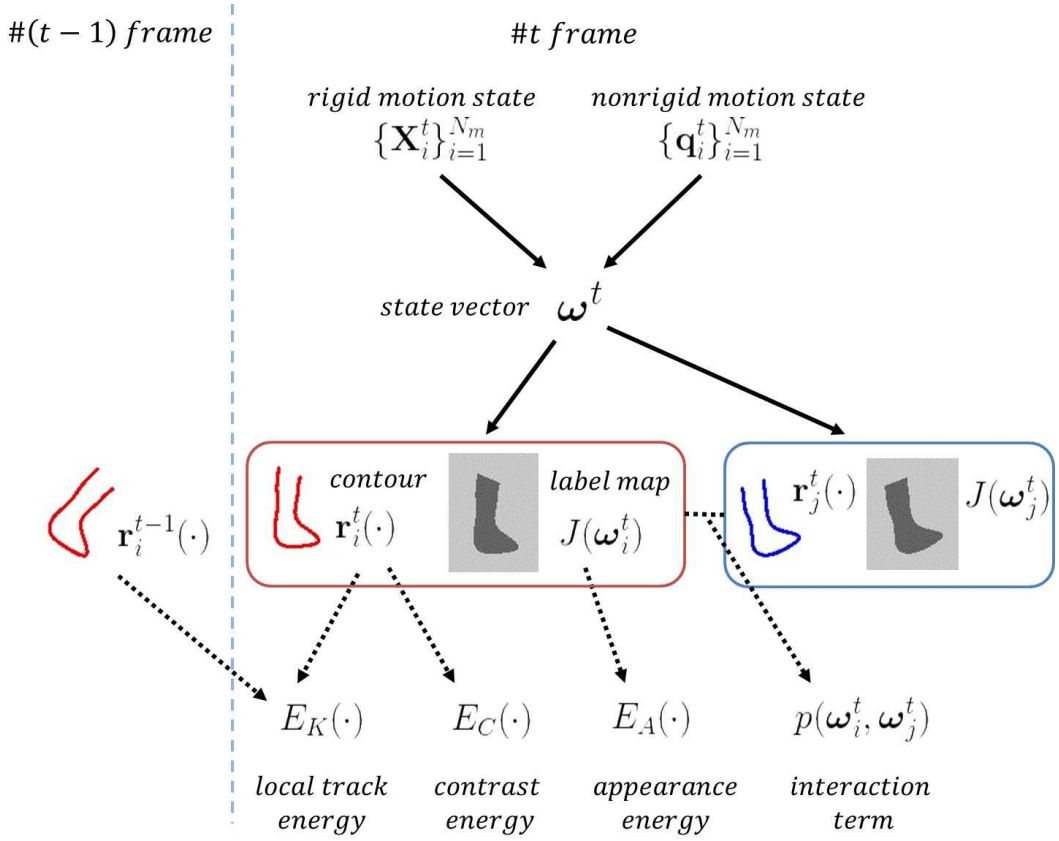


Figure 1.3: Flowchart for multi-contours tracking. The state vector $\boldsymbol{\omega}^t$ consists of N_m contours that the rigid motion state $\mathbf{X}_i^t$ and nonrigid motion state $\mathbf{q}_i^t$ are notations of i-th contour. The contour $\mathbf{r}_i^t(\cdot)$ is derived and measured by E_C, E_K and E_A, moreover, the interaction term between contours $\mathbf{r}_i^t$ and $\mathbf{r}_j^t$ is added.

tions close to the estimated position with a uniform motion. Moreover, the proposed method draws particle samples with a new manner that several combinations of contours should be from each superior particle set Ψ_i^t that will be explained in Section 5.1.3 and other particles are selected at random. This sampling way is designed for drawing good particles efficiently while the tracker maintains the randomness of choosing particles.

1.4 Structure of the dissertation

The rest of this dissertation is organized as follows. Chapter 2 reviews three topics related to the proposed method. The framework for open contour tracking is provided in Chapter 3 where the proposed shape model is formulated dividing rigid and non-rigid motions and the training criterion for non-rigid motions is also developed. Next, Chapter 4 presents the closed contour tracking with the same framework except that local track term is replaced newly and B-spline models are changed. Moreover, the automatic document capture system is also proposed that a document shape is estimated automatically and efficiently with well-known its quadrilateral shape. Next, multi-contours tracking is conducted in Chapter 5. The state space is expanded for multi-contours and a new interaction term between contours is adopted to consider the interactions. By this term, targets that belong in the same class can be distinguished and estimated precisely. Finally, this dissertation is concluded in Chapter 6.

Chapter 2

Related work

Numerous methods have been proposed for object tracking and a comprehensive review is beyond the scope of this dissertation (interested readers may refer to [1]). Rather, in this chapter, we review three related topics: bounding box tracking, contour tracking and multi-objects tracking.

2.1 Bounding box tracking

Tracking methods can be classified into several categories according to their object representation methods (e.g., bounding boxes, points, geometric shapes, and contours), where most recent methods use the bounding box representation. For example, in [2], the mean shape and the basis describing target variations were used for the tracking and the authors developed its incremental update method. The tracking method in [8] was also based on this generative framework, where they improved the filtering performance by adopting geometrically derived models. In contrast to the generative framework, which learns the characteristics of the target, the discrim-

inative approach attempts to address the tracking problem by learning the difference between the target and its surrounding background [4–7]. For instance, the authors in [6] achieved the goal through a multiple instance learning, and a long term tracker was developed by using a P-N learning method in [7]. In [4], a very efficient tracker was proposed by exploiting the circulant matrix structure of the training samples. The tracker in [5] was based on compressive sensing theory, so that it used very low dimensional features. Although many methods showed impressive results, bounding boxes inevitably contain background regions, and thus bounding box based approach usually suffers from the drift problem. To address this problem, several methods attempted to minimize the effect of backgrounds by segmenting out the background regions in bounding boxes: they assigned background labels to pixels [3], or to the superpixels [9, 10] or patches [11, 12]. These methods improved the tracking performances, however, their goal was not object segmentation and hence accurate object boundaries were not available.

2.2 Contour tracking

Because contour-based tracking provides boundary information as well as location information, it is considered an effective way to handle deformable objects in video processing. For example, in [13, 14, 24–27], contours were represented with level-set functions, and tracking was realized through the evolution of these level-set functions. Although the level-set-based approach provides a mathematically sound framework, the consideration of appearance information in this framework is rather limited and the method can handle only closed contours. The contours can also be represented with segmentation masks. For instance, local object/background

classifiers were trained for data terms and masks were estimated with the graph-cut method in [16]. Similarly, in [17], the object location was determined with the Hough voting technique and a mask was obtained with the Grabcut algorithm [28]. However, imposing shape constraints in this approach is not simple. Unlike the foregoing implicit representation methods, contours can be explicitly represented with active contour models or point sets. In [18,19], open contour tracking methods were proposed, which represent contours with snakes and sets of points respectively. However, they developed algorithms for very specific targets: partial tongue structures in ultrasound images [18] and microtubules in biological image fields [19]. In the seminal work of Isard and Blake [15], contours were represented with B-splines so that closed and open contours were handled in the same framework. However, in the work, a very simple measurement term was adopted and the method had difficulties in handling complex sequences (e.g., having background clutters). In order to improve the measurement term, numerous methods have been proposed by exploiting various features: color, intensity or contrast response were considered in [13, 14, 16, 19] by using histogram representation and Gaussian mixture models; the local intensity difference on both sides of the boundary was exploited in [18]; and features in local patches were used to improve the tracking performance for nonrigid deformation and occlusions [17]. Although they could improve tracking performance, it is not straightforward to apply these measurement terms to the contour tracking problem when the contour is open.

2.3 Multi-objects tracking

Most conventional tracking methods for multi-objects assumed that they already had the results of object detector every frames and solved the data association problem [29–32]. To be precise, the results of object detector were linked the best matching one between frames to construct a tracklet which was a set of tracking trajectory for each target. Then, they connected tracklets to make whole tracking trajectories. Therefore, they usually selected people as their tracking targets and pedestrians are detected in advance using various human detectors [33–35]. In [35], they detected not only the location of human but also their parts and poses. On the other hand, multi-objects whose model were constructed beforehand were tracked based on Markov chain Monte Carlo (MCMC) method [36, 37]. In [36], all targets were represented by one state vector and their positions were estimated with learned color model of targets. Also, they accepted a variable number of targets using reversible-jump MCMC (RJMCMC). In [37], authors set people to their targets and multi-objects tracking and segmentation were carried out with a unified framework. However, the camera should be fixed, and models of the camera and targets are essential to run the algorithm.

Chapter 3

Open contour tracking based on a nonrigid shape training

3.1 Proposed state space model

In this section, we explain the proposed state space model consisting of a state vector $\boldsymbol{\omega}^t$ that represents the contour of an object and its stochastic dynamic model $p(\boldsymbol{\omega}^t|\boldsymbol{\omega}^{t-1})$.

3.1.1 Reviews on the active contour model

For the representation of open contours, we adopt a B-spline model that represents curves with a linear combinations of basis functions [38]. Let us denote the vector consisting of n basis functions as

$$\mathbf{B}(u) = (B_1(u), B_2(u), \cdots, B_n(u))^\top \in \mathbb{R}^n, \tag{3.1}$$

then the contour $\mathbf{r}(u)$ on $u \in [0, L]$ (u is a continuous real variable) is given by

$$\begin{aligned} \mathbf{r}(u) &= (x(u), y(u))^\top \\ &= (\mathbf{B}(u)^\top \mathbf{D}^\top)^\top \in \mathbb{R}^2 \end{aligned} \tag{3.2}$$

where $\mathbf{D}$ is a $2 \times n$ matrix consisting of control points that determine the weights of the basis functions. In all experiments, we adopt aperiodic cubic B-splines, so that $L = n - 3$. As an example, we show objects, their corresponding contours and control points ($\mathbf{r}(\cdot)$ and $\mathbf{D}$) in Fig. 3.1.

3.1.2 Proposed state vector

Because we adopt B-splines, we need to describe the contour in the t-th frame with n control points $\mathbf{D}^t$. To achieve this goal, we assume that the nonrigid motion is applied to the mean shape $\bar{\mathbf{D}}$ and then this deformed shape experiences rigid motions, yielding $\mathbf{D}^t$.

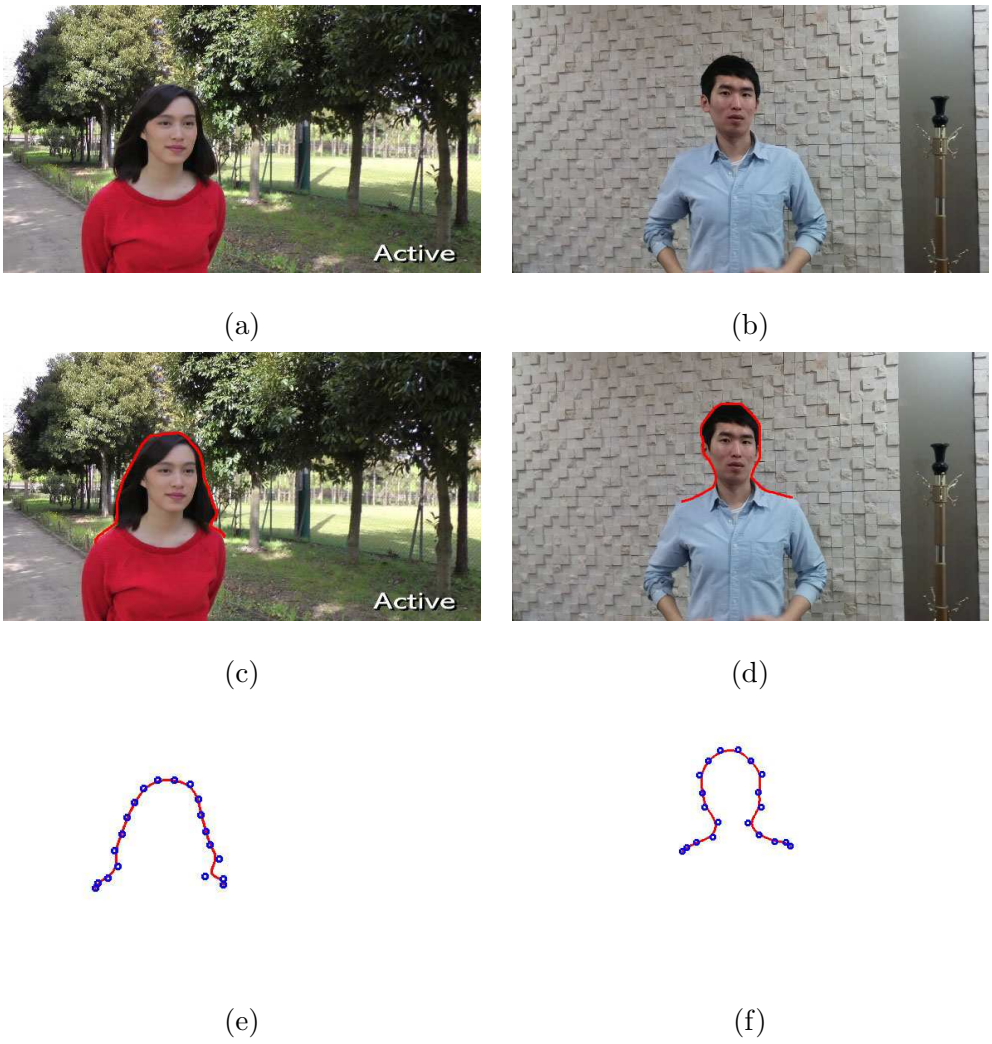
To be precise, for the description of nonrigid motion, we find the basis vectors of the nonrigid motion and describe the motion with a linear combination of them [15]:

$$\bar{\mathbf{D}} + \text{reshape}(\mathbf{W}\mathbf{q}^t) \tag{3.3}$$

where $\mathbf{W} \in \mathbb{R}^{2n \times k}$ is an orthonormal matrix whose columns are k nonrigid shape basis, $\mathbf{q}^t \in \mathbb{R}^{k \times 1}$ is a nonrigid weight vector, and reshape$(\cdot)$ is a rearrangement operator:

$$\begin{aligned} &\text{reshape}((x_1, \cdots, x_n, y_1, \cdots, y_n)^\top) \\ &\qquad = \begin{pmatrix} x_1 & \cdots & x_n \\ y_1 & \cdots & y_n \end{pmatrix}. \end{aligned} \tag{3.4}$$

We consider $\mathbf{q}^t$ as the nonrigid state (as discussed in Section 1.1), because this vector determines the nonrigid shape variations from the mean shape. Then, the control



(a) (b)

(c) (d)

(e) (f)

Figure 3.1: First row: input images. Second row: ground truth contours. Third row: extracted contours $\mathbf{r}(u)$ (red curve) and their control points $\mathbf{D}$ (blue circles).

points of the t-th frame are given by the rigid transformation of (3.3):

$$\mathbf{D}^t = s^t \mathbf{R}^t (\bar{\mathbf{D}} + \text{reshape}(\mathbf{W}\mathbf{q}^t)) + \mathbf{p}^t \mathbf{1}_{n\times 1}^\top. \tag{3.5}$$

where s^t is a scale, $\mathbf{p}^t \in \mathbb{R}^{2\times 1}$ is a translation vector, $\mathbf{1}_{n\times 1}$ is a column vector consisting of 1's, and $\mathbf{R}^t$ is a rotation matrix. Therefore, we can represent the rigid motion state with a 3×3 matrix:

$$\mathbf{X}^t = \begin{pmatrix} s^t\mathbf{R}^t & \mathbf{p}^t \\ \mathbf{0}_{1\times 2} & 1. \end{pmatrix} \tag{3.6}$$

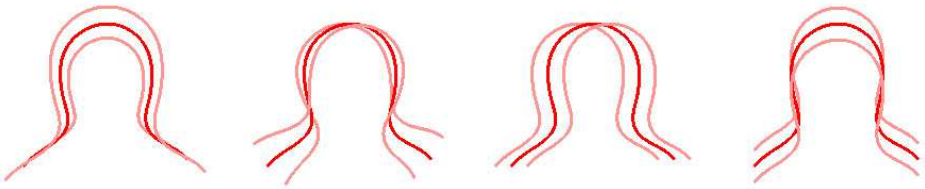
Finally, the state vector $\boldsymbol{\omega}^t$ at the t-th frame consists of $\mathbf{X}^t$ and $\mathbf{q}^t$, whose degree of freedom is $k + 4$.

3.1.3 Proposed stochastic dynamic model

Let us assume that we are building an open-contour tracker for a given scenario. When implementing the conventional approach [15] for this, we have to collect samples showing possible rigid/nonrigid motions (and their combinations) in the scenario, and train the stochastic dynamic model $p(\boldsymbol{\omega}^t|\boldsymbol{\omega}^{t-1})$ from them. However, collecting samples for all possible combinations is not a simple task, and moreover, the training result is not transferable. In other words, we cannot transfer one training result (trained on videos having little rotations) to other scenarios (that may require rotations). In order to alleviate these limitations, we propose a separable stochastic dynamic model:

$$p(\boldsymbol{\omega}^t|\boldsymbol{\omega}^{t-1}) = p(\mathbf{X}^t|\mathbf{X}^{t-1})\ p(\mathbf{q}^t|\mathbf{q}^{t-1}). \tag{3.7}$$

This approach decouples nonrigid motions from rigid motions and we can focus on the training of the nonrigid motions by adopting geometrically well-defined models for rigid motion dynamics. In particular for the rigid motion, we approximate



(a) $\mathbf{E}_1$ (b) $\mathbf{E}_2$ (c) $\mathbf{E}_3$ (d) $\mathbf{E}_4$

Figure 3.2: Rigid motions induced by each of the basis vectors: (a) scaling, (b) rotation, (c) horizontal translation, (d) vertical translation.

$p(\mathbf{X}^t|\mathbf{X}^{t-1})$ with

$$N_{\mathit{Aff}(2)}(\mathbf{X}^{t-1}, \mathbf{S}), \tag{3.8}$$

based on [8], where $N_{\mathit{Aff}(2)}(\mathbf{X}, \mathbf{S})$ is an approximated normal distribution on $\mathit{Aff}(2)$ whose center is $\mathbf{X} \in \mathit{Aff}(2)$ and $\mathbf{S}$ is a covariance matrix on $\mathit{aff}(2)$ that is a tangential space corresponding to the manifold $\mathit{Aff}(2)$ (the explicit representation of $\mathit{Aff}(2)$ and $\mathit{aff}(2)$ can be found in [8]). In other words, (3.8) means

$$\mathbf{X}^t \simeq \mathbf{X}^{t-1} \cdot \exp\left(\sum_{i=1}^{4} e_i \mathbf{E}_i\right) \tag{3.9}$$

where $(e_1, \cdots, e_4)$ is a zero-mean Gaussian vector whose covariance matrix is $\mathbf{S}$, and $\mathbf{E}_i$ is a basis of $\mathit{aff}(2)$ (for the similarity transformation):

$$\begin{aligned} \mathbf{E}_1 &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{E}_2 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \\ \mathbf{E}_3 &= \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{E}_4 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}. \end{aligned} \tag{3.10}$$

As an example, rigid motions induced by these basis vectors are illustrated in Fig. 3.2.

For the modeling of nonrigid transformations, we assume that random variables in

$$\mathbf{q}^t = (q_1^t, q_2^t, \ldots, q_k^t) \tag{3.11}$$

are independent and its distribution is given by the product of Gaussian and Laplace distributions:

$$p(\mathbf{q}^t \mid \mathbf{q}^{t-1}) = \prod_i N(q_i^t; q_i^{t-1}, \sigma_i^2) \times \frac{\lambda_R}{2} \exp(-\lambda_R |q_i^t|), \tag{3.12}$$

where $N(\cdot; \mu, \sigma^2)$ is a Gaussian distribution with the mean μ and standard deviation σ. The Laplace distribution plays the role of a regularization term that penalizes overfitting: the distribution prefers a small $|q_i^t|$ and prevents $\mathbf{W}\mathbf{q}^t$ from being arbitrarily large, i.e., the shape should be similar to the mean shape.

3.2 Training of the proposed state space model

In this section, we present a training criterion and optimization method for the proposed state space model. For training samples, we manually draw object boundaries for collected images and compute their control points by fitting them to B-splines as shown in Fig. 3.1. In order to facilitate the training, we remove translations and normalize scales in training samples, and build a set of control points

$$\mathcal{D} = \{\mathbf{D}_i\}\,. \tag{3.13}$$

Algorithm 1 Proposed training algorithm. Note that we use a vectorization function vec(·), which is the inverse of a reshape(·) function.

Input: $\mathcal{D} = \{\mathbf{D}_1, \cdots, \mathbf{D}_{N_d}\}$

Output: $\bar{\mathbf{D}}, \mathbf{W}, \{\sigma_1, \cdots, \sigma_k\}$

for $i = 1 \cdots N_d$ **do**

  Set $s_i = 1, \mathbf{R}_i = \mathbf{I}, \mathbf{q}_i = \mathbf{0}$ ▷ Initialization

end for

repeat

  repeat

    $\bar{\mathbf{D}} = \sum_{i=1}^{N_d} [\frac{1}{s_i} \mathbf{R}_i^\top \mathbf{D}_i - \text{reshape}(\mathbf{W}\mathbf{q}_i)]$

    $\bar{\mathbf{D}} \leftarrow \frac{\bar{\mathbf{D}}}{\|\bar{\mathbf{D}}\|_F}$

    for $i = 1 \cdots N_d$ **do**

      $\mathbf{D}_i(\bar{\mathbf{D}} + \text{reshape}(\mathbf{W}\mathbf{q}_i))^\top = \mathbf{U}_i \mathbf{\Sigma}_i \mathbf{V}_i^\top$

      ▷ Singular value decomposition (SVD)

      $\mathbf{R}_i = \mathbf{U}_i \mathbf{V}_i^\top$

      $s_i = \frac{1}{\text{tr}(\mathbf{\Sigma}_i)}$

    end for

  until convergence

Algorithm 1 (continued)

for $i = 1 \cdots N_d$ **do**

  $\mathbf{z}_i = (\frac{1}{s_i}\mathbf{R}_i^\top \mathbf{D}_i - \bar{\mathbf{D}})$

  $\mathbf{z}_i \leftarrow (\mathbf{I} - \mathbf{P}\mathbf{P}^\top)\text{vec}(\mathbf{z}_i)$

end for

$\mathbf{W} = \text{PCA}(\{\mathbf{z}_i\}_{i=1}^{N_d})$

for $i = 1 \cdots N_d$ **do**

  $\mathbf{q}_i = \mathbf{W}^\top \mathbf{z}_i$

end for

until convergence

$\{\sigma_j^2\}_{j=1}^k = \text{Var}(\mathbf{q}_i)_{i=1}^{N_d}$ ▷ Element-wise variance

3.2.1 Training criterion

For the learning of parameters in (3.5), we solve the Frobenius-norm optimization problem

$$\underset{\{s_i,\mathbf{R}_i,\mathbf{q}_i\},\bar{\mathbf{D}},\mathbf{W}}{\arg\min} \sum_i \|\epsilon_i\|_F^2 \tag{3.14}$$

subject to

$$\mathbf{D}_i = s_i\mathbf{R}_i(\bar{\mathbf{D}} + \text{reshape}(\mathbf{W}\mathbf{q}_i) + \epsilon_i) \tag{3.15}$$

and the following 5 additional constraints (we have 6 constraints in (3.15) $\sim$ (3.20) in the optimization). Intuitively, ϵ_i measures the difference between true control points $\mathbf{D}_i$ and our representation using a mean shape and a small number of basis vectors ($\bar{\mathbf{D}} + \text{reshape}(\mathbf{W}\mathbf{q}_i)$) when rigid motions are compensated. We also impose

constraints on the scale, rotation and center of the mean shape:

$$\|\bar{\mathbf{D}}\|_F = \quad 1, \tag{3.16}$$

$$\mathbf{R}_i^\top \mathbf{R}_i = \quad \mathbf{I}_{2\times 2}, \tag{3.17}$$

$$\bar{\mathbf{D}}\mathbf{1}_{n\times 1} = \quad \mathbf{0}_{2\times 1}. \tag{3.18}$$

Finally, because $\mathbf{W}$ is a nonrigid basis matrix, we impose the constraints that column vectors are orthonormal and orthogonal to rigid motion:

$$\mathbf{W}^\top \mathbf{W} = \quad \mathbf{I}_{k\times k} \tag{3.19}$$

$$\mathbf{P}^\top \mathbf{W} = \quad \mathbf{0}_{4\times k}. \tag{3.20}$$

Here, $\mathbf{P}$ is given by

$$\mathbf{P} = \begin{pmatrix} \begin{pmatrix} \mathbf{1}_{n\times 1} \\ \mathbf{0}_{n\times 1} \end{pmatrix} & \begin{pmatrix} \mathbf{0}_{n\times 1} \\ \mathbf{1}_{n\times 1} \end{pmatrix} & \begin{pmatrix} \bar{\mathbf{D}}^x \\ \bar{\mathbf{D}}^y \end{pmatrix} & \begin{pmatrix} -\bar{\mathbf{D}}^y \\ \bar{\mathbf{D}}^x \end{pmatrix} \end{pmatrix} \tag{3.21}$$

where $\bar{\mathbf{D}}^x, \bar{\mathbf{D}}^y \in \mathbb{R}^{n\times 1}$ are x and y-space components of $\bar{\mathbf{D}}$ respectively.

3.2.2 Optimization method

In order to solve this optimization problem, we develop an iterative method, which is summarized in Algorithm 1. The algorithm consists of two steps. In the first step, we estimate $(\{s_i\}, \{\mathbf{R}_i\}, \bar{\mathbf{D}})$ while fixing $(\mathbf{W}, \{\mathbf{q}_i\})$, which is similar to the generalized Procrustes analysis (GPA) [39]. The other step is to estimate $(\mathbf{W}, \{\mathbf{q}_i\})$ while fixing other parameters. This step is also similar to the principal component analysis (PCA) except that we remove rigid components before PCA (line 16 in Algorithm 1) in order to impose the constraint in (3.20). We also provide proofs for each variable in Section 3.2.3. Finally, given the nonrigid state vectors of all

samples, we train parameters in the dynamic model in (3.12). Note that we do not need the training of rigid dynamics.

3.2.3 Proof for solving the training problem

We solve the training problem (3.14) with constraints (3.15) $\sim$ (3.20) using iteration that only one variable is optimized during fixing others. Here provides the proof for each variable in Algorithm 1.

Optimize $\bar{\mathbf{D}}$

$$\begin{aligned}\sum_i \|\epsilon_i\|_F^2 &= \sum_i \|\frac{1}{s_i}\mathbf{R}_i^\top \mathbf{D}_i - \text{reshape}(\mathbf{W}\mathbf{q}_i) - \bar{\mathbf{D}}\|_F^2 \\ &= \sum_i \|\mathbf{a}_i - \bar{\mathbf{D}}\|_F^2 = \sum_i \|\mathbf{a}_{i,v} - \bar{\mathbf{D}}_v\|^2 \\ &= \sum_i [\|\mathbf{a}_{i,v}\|^2 + \|\bar{\mathbf{D}}_v\|^2 - 2\bar{\mathbf{D}}_v^\top \mathbf{b}_{i,v}] \\ &= -2\sum_i \bar{\mathbf{D}}_v^\top \mathbf{a}_{i,v} + C = -2\bar{\mathbf{D}}_v^\top \sum_i \mathbf{a}_{i,v} + C\end{aligned} \tag{3.22}$$

where $\mathbf{a}_i = \frac{1}{s_i}\mathbf{R}_i^\top \mathbf{D}_i - \text{reshape}(\mathbf{W}\mathbf{q}_i)$, $\mathbf{a}_{i,v}$ and $\bar{\mathbf{D}}_v$ stand for the vectorization results of $\mathbf{a}_i$ and $\bar{\mathbf{D}}$ respectively, and C is a constant. Then, $\bar{\mathbf{D}}$ is selected to minimize the cost:

$$\therefore \bar{\mathbf{D}} = \frac{\sum_i \mathbf{a}_i}{\|\sum_i \mathbf{a}_i\|} \tag{3.23}$$

(line 6 and 7 in Algorithm 1) and $\bar{\mathbf{D}}$ satisfies following constraints:

$$\bar{\mathbf{D}}\mathbf{1}_{n\times 1} = \mathbf{0}_{2\times 1} \tag{3.24}$$

$$\|\bar{\mathbf{D}}\|_F = 1. \tag{3.25}$$

Optimize $\mathbf{R}_i$

$$\begin{aligned}
\|\epsilon_i\|_F^2 =& \|\frac{1}{s_i}\mathbf{R}_i^\top \mathbf{D}_i - (\text{reshape}(\mathbf{W}\mathbf{q}_i) + \bar{\mathbf{D}})\|_F^2 \\
=& \|\frac{1}{s_i}\mathbf{R}_i^\top \mathbf{D}_i - \mathbf{b}_i\|_F^2 \\
=& \text{tr}\left[\mathbf{D}_i^\top \mathbf{R}_i \frac{1}{s_i^2}\mathbf{R}_i^\top \mathbf{D}_i - 2\mathbf{b}_i^\top \frac{1}{s_i}\mathbf{R}_i^\top \mathbf{D}_i + \mathbf{b}_i^\top \mathbf{b}_i\right] \\
=& \frac{-2}{s_i}\text{tr}(\mathbf{b}_i^\top \mathbf{R}_i^\top \mathbf{D}_i) + K = \frac{-2}{s_i}\text{tr}(\mathbf{R}_i^\top \mathbf{D}_i \mathbf{b}_i^\top) + K \\
& (\because \frac{1}{s_i}\mathbf{D}_i \mathbf{d}_i^\top = \mathbf{U}_i \mathbf{\Sigma}_i \mathbf{V}_i^\top \quad \text{by SVD}) \\
=& \frac{-2}{s_i}\text{tr}(\mathbf{R}_i^\top \mathbf{U}_i \mathbf{\Sigma}_i \mathbf{V}_i^\top) + K = \frac{-2}{s_i}\text{tr}(\mathbf{V}_i^\top \mathbf{R}_i^\top \mathbf{U}_i \mathbf{\Sigma}_i) + K \\
=& \frac{-2}{s_i}\text{tr}(\mathbf{M}_i \mathbf{\Sigma}_i) + K = \frac{-2}{s_i}(m_{11}\sigma_1 + m_{22}\sigma_2) + K \\
\geq& \frac{-2}{s_i}(\sigma_1 + \sigma_2) + K
\end{aligned} \tag{3.26}$$

where $\mathbf{b}_i = \text{reshape}(\mathbf{W}\mathbf{q}_i) + \bar{\mathbf{D}}$ and $\mathbf{M}_i = \mathbf{V}_i^\top \mathbf{R}_i^\top \mathbf{U}_i$ that is orthonormal. Since the cost should be minimized, $\mathbf{M}_i$ is chosen as an identity matrix (equality condition in the last row in (3.26)):

$$\begin{aligned}
& \therefore \mathbf{V}_i^\top \mathbf{R}_i^\top \mathbf{U}_i = \mathbf{M}_i = \mathbf{I}_{2\times 2} \\
& \Leftrightarrow \mathbf{R}_i = \mathbf{U}_i \mathbf{V}_i^\top
\end{aligned} \tag{3.27}$$

(line 10 in Algorithm 1) and $\mathbf{R}_i$ satisfies the constraint $\mathbf{R}_i^\top \mathbf{R}_i = \mathbf{I}_{2\times 2}$

Optimize s_i

$$\begin{aligned}
\|\epsilon_i\|_F^2 =& \text{tr}\left[\mathbf{D}_i^\top \mathbf{R}_i \frac{1}{s_i^2}\mathbf{R}_i^\top \mathbf{D}_i - 2\mathbf{b}_i^\top \frac{1}{s_i}\mathbf{R}_i^\top \mathbf{D}_i + \mathbf{b}_i^\top \mathbf{b}_i\right] \\
=& \frac{1}{s_i^2} - 2\text{tr}(\mathbf{b}_i^\top \mathbf{R}_i^\top \mathbf{D}_i)\frac{1}{s_i} + K.
\end{aligned} \tag{3.28}$$

The cost is minimized when $\frac{1}{s_i}$ is set to the critical point of a quadratic function:

$$\begin{aligned}
& \frac{1}{s_i} = \text{tr}(\mathbf{b}_i^\top \mathbf{R}_i^\top \mathbf{D}_i) = \text{tr}(\mathbf{R}_i^\top \mathbf{D}_i \mathbf{b}_i^\top) = \text{tr}(\mathbf{V}_i \mathbf{U}_i^\top \mathbf{U}_i \mathbf{\Sigma}_i \mathbf{V}_i^\top) = \text{tr}(\mathbf{\Sigma}_i) \\
& \Leftrightarrow s_i = \frac{1}{\text{tr}(\mathbf{\Sigma}_i)}
\end{aligned} \tag{3.29}$$

which is same to the line 11 in Algorithm 1.

Optimize $\mathbf{W}, \mathbf{q}_i$

$$
\begin{aligned}
&\arg\min_{\mathbf{W},\mathbf{q}_i} \sum_i \|\epsilon_i\|_F^2 \\
&= \arg\min \sum_i \|\mathrm{vec}(\frac{1}{s_i}\mathbf{R}_i^\top \mathbf{D}_i - \bar{\mathbf{D}}) - \mathbf{W}\mathbf{q}_i\|^2 \\
&= \arg\min \sum_i \|\mathbf{y}_i - \mathbf{W}\mathbf{q}_i\|^2 \\
&= \arg\min \sum_i \|\mathbf{W}\mathbf{q}_i - (\mathbf{I} - \mathbf{P}\mathbf{P}^\top)\mathbf{y}_i - \mathbf{P}\mathbf{P}^\top \mathbf{y}_i\|^2 \\
&= \arg\min \sum_i [\|\mathbf{W}\mathbf{q}_i - (\mathbf{I} - \mathbf{P}\mathbf{P}^\top)\mathbf{y}_i\|^2 + \|\mathbf{P}\mathbf{P}^\top \mathbf{y}_i\|^2] \\
&\quad (\because \text{orthogonality}) \\
&= \arg\min \sum_i \|\mathbf{W}\mathbf{q}_i - \mathbf{z}_i\|^2 \\
&= \arg\min \sum_i (\mathbf{q}_i^\top \mathbf{q}_i - 2\mathbf{z}_i^\top \mathbf{W}\mathbf{q}_i)
\end{aligned}
\tag{3.30}
$$

where $\mathbf{y}_i = \mathrm{vec}(\frac{1}{s_i}\mathbf{R}_i^\top \mathbf{D}_i - \bar{\mathbf{D}})$ and $\mathbf{z}_i = (\mathbf{I} - \mathbf{P}\mathbf{P}^\top)\mathbf{y}_i$ and $\mathbf{P}^\top \mathbf{z}_i = \mathbf{0}_{4\times 1}$. Since partial derivative by $\mathbf{q}_i$ is zero,

$$
\therefore \mathbf{q}_i = \mathbf{W}^\top \mathbf{z}_i \tag{3.31}
$$

(line 20 in Algorithm 1) and we can eliminate $\mathbf{q}_i$ from the object function as

$$
\begin{aligned}
&\arg\min_{\mathbf{W}} \sum_i (\mathbf{z}_i^\top \mathbf{W}\mathbf{W}^\top \mathbf{z}_i - 2\mathbf{z}_i^\top \mathbf{W}\mathbf{W}^\top \mathbf{z}_i) \\
&= \arg\max \sum_i \mathbf{z}_i^\top \mathbf{W}\mathbf{W}^\top \mathbf{z}_i
\end{aligned}
\tag{3.32}
$$

with constraint $\mathbf{W}^\top \mathbf{W} = \mathbf{I}$. By Lagrange multiplier and partial derivative by a column vector of $\mathbf{W}$,

$$
\sum_i \mathbf{z}_i \mathbf{z}_i^\top \mathbf{w}_j = \lambda_j \mathbf{w}_j \tag{3.33}
$$

where $\mathbf{w}_j$ is j-th column of $\mathbf{W}$. Therefore, we derive $\mathbf{W}$ as the set of eigen-vectors of $\sum_i \mathbf{z}_i \mathbf{z}_i^\top$ (line 18 in Algorithm 1).

3.3 Measurement

Given the proposed state space model, we can explain the dynamics of the target. However, for robust tracking, we also develop a measurement function that exploits a variety of cues in videos. To this end, we propose a measurement function that considers three complementary factors:

$$p(I^t|\boldsymbol{\omega}^t) \propto \exp\left(-\lambda_C E_C(\boldsymbol{\omega}^t) - \lambda_K E_K(\boldsymbol{\omega}^t) - \lambda_A E_A(\boldsymbol{\omega}^t)\right) \tag{3.34}$$

where E_C considers contrast, the local track term E_K reflects temporal coherence, and E_A is an appearance term (colors in the region).

3.3.1 Contrast term

Because object boundaries usually show intensity discontinuities (edges), the contrast term becomes small when the boundaries are on the edges. However, since there are also a lot of high contrast regions (e.g., textures) other than the object boundary, we remove textures by using a domain transform filter [40]. An input image and its filtered image by the domain transfrom filter $F(\cdot)$ are shown in Fig. 3.3. The contrast energy is given by

$$E_C(\boldsymbol{\omega}^t) \propto \sum_{\mathbf{x}\in\mathbf{r}^t(u)} \frac{1}{1 + |\nabla(F \circ I^t)(\mathbf{x}))|^2}. \tag{3.35}$$

where $(F \circ I^t)$ is the filtering result of I^t and the contour $\mathbf{r}^t(u)$ is derived from the state $\boldsymbol{\omega}^t$. As shown in Fig. 3.3, we reduce the effects of textures by the filtering and $E_C(\boldsymbol{\omega}^t)$ becomes small for true object boundaries.

(a) (b)

(c) (d)

Figure 3.3: (a) Input image I^t, (b) smoothed image $(F \circ I^t)$, (c) inverse contrast image $\frac{1}{1+|\nabla(F \circ I^t)(\mathbf{x}))|^2}$, (d) open contour $\mathbf{r}^t(u)$ (red curve) and control points $\mathbf{D}^t$ (blue circles).

(a) (b)

Figure 3.4: Illustration of (a) $J(\boldsymbol{\omega}^t)$ and (b) $J(I^t)$ for a human omega shape model (different labels are coded with different intensities). $J(\boldsymbol{\omega}^t)$ is derived from $\boldsymbol{\omega}^t$ and $J(I^t)$ is the per-pixel classification result.

3.3.2 Local track term

In order to consider temporal coherence, we first select knot points (in other words, $\mathbf{r}^{t-1}(u)$ for $u = 0, 1, \ldots, L$) in the $(t-1)$-th frame. Then, we find their corresponding points in the t-th frame by using the sum of absolute differences (SAD) criterion. We denote the corresponding point of $\mathbf{r}^{t-1}(u)$ as $\tilde{\mathbf{r}}^t(u)$, and the local track energy is given by

$$E_K(\boldsymbol{\omega}^t) \propto \sum_{u=0}^{L} \min(\beta \left\|\mathbf{r}^t(u) - \tilde{\mathbf{r}}^t(u)\right\|^2, 1). \tag{3.36}$$

This term penalizes the large changes in knot points, however, we employ the truncated square functions in order to allow abrupt motions.

3.3.3 Appearance term

For the appearance term, we build two label maps where each pixel in the map represents the label $l \in \mathcal{L}$, where $\mathcal{L}$ is a set of labels. The first map is denoted $J(\boldsymbol{\omega}^t)$ and it is derived from the current state $\boldsymbol{\omega}^t$ as shown in Fig. 3.4-(a) (the number

(a) (b) (c)

Figure 3.5: Illutrations of (a) $p^t_{\mathbf{x}}(\text{“head”})$, (b) $p^t_{\mathbf{x}}(\text{“body”})$ and (c) $p^t_{\mathbf{x}}(\text{“background”})$ for a human omega shape. The intensity means the probability value: $\sum_{l\in\mathcal{L}} p^t_{\mathbf{x}}(l) = 1$.

of labels and the construction of $J(\boldsymbol{\omega}^t)$ from $\boldsymbol{\omega^t}$ is application-dependent, and this will be further discussed in the experimental section). The other label map is called $J(I^t)$ and it is a pixelwise classification result as shown in Fig. 3.4-(b). To be precise, for the pixelwise classification, we keep track of two probabilistic models:

- the color distribution of each label $p^t(\cdot|l)$,
- the prior probability of each label at a point $\mathbf{x}$ $p^t_{\mathbf{x}}(l)$.

Using the probabilistic models, the label at the t-th frame is determined by

$$l^t_{\mathbf{x}} = \arg\max_{l\in\mathcal{L}} p^t((F \circ I^t)(\mathbf{x}))|l)p^t_{\mathbf{x}}(l). \tag{3.37}$$

Here, the point $\mathbf{x}$ is determined by reflecting a displacement vector from the center of the control points in the previous frame in order to compensate for the object motion. In particular, $p^t(\cdot|l)$ is the color distribution of pixels in each region, and we adopt a $32 \times 32 \times 32$ RGB color histogram for each label, in other words, $|\mathcal{L}| - 1$ histograms are used (“don't care” label is not considered). For the prior probability

$p^t_{\mathbf{x}}(l)$, we use a histogram having $|\mathcal{L}| - 1$ bins for each displacement vector $\mathbf{x}$. Fig. 3.5 illustrates $p^t_{\mathbf{x}}(l)$ around the target object. Finally, the appearance term is given by

$$E_A(\boldsymbol{\omega}^t) \propto \|J(\boldsymbol{\omega}^t) - J(I^t)\|_0. \tag{3.38}$$

As shown in Fig. 3.4-(a), $J(\boldsymbol{\omega}^t)$ has "don't care" regions (black pixels) that are not considered in the evaluation of $E_A(\boldsymbol{\omega}^t)$.

3.3.4 Model update

On the basis of the foregoing energy terms, we can estimate the optimal state $\hat{\boldsymbol{\omega}}^t$ in the current frame. Using the estimator, we update two probabilistic models used in the appearance term:

$$p^{t+1}(\cdot|l) = (1-\gamma)p^t(\cdot|l) + \gamma c^t(\cdot|l) \tag{3.39}$$

$$p^{t+1}_{\mathbf{x}}(l) = (1-\gamma)p^t_{\mathbf{x}}(l) + \gamma h^t_{\mathbf{x}}(l) \tag{3.40}$$

where $c^t(\cdot|l)$ is a $32 \times 32 \times 32$ RGB-color histogram of the pixels having a label $l \in \mathcal{L}$ and $h^t_{\mathbf{x}}(l)$ is the label histogram around $\mathbf{x}$ at the t-th frame. For the robustness, we apply Gaussian smoothing to the color histogram and add a constant to the label histogram.

3.3.5 Weights of three measurement terms

The proposed measurement function consists of three complementary terms and we need to determine their weights. Experimental results show that a fixed setting, i.e., $\lambda_C = 10$, $\lambda_K = 2.5$, $\lambda_A = 50$, works well in practice. However, the performance can be further improved by employing an adaptive scheme that puts more weights on

more reliable terms. In particular, we represent each weight with a linear function of an indicating feature:

$$\lambda_i = a_i \mu_i + b_i \tag{3.41}$$

where $i \in \{C, K, A\}$. The weight of the contrast term (λ_C) should decrease when there are background clutters, and we set μ_C to the mean contrast around the target:

$$\mu_C = \frac{1}{|N(\mathbf{r}^{t-1}(u))|} \sum_{\mathbf{x} \in N(\mathbf{r}^{t-1}(u))} \left| \nabla (F \circ I^t)(\mathbf{x})) \right| \tag{3.42}$$

where $N(\mathbf{r}^{t-1}(u))$ is a bounding rectangle of $\mathbf{r}^{t-1}(u)$, similar to the rectangle in Fig. 3.8-(a) and $|N(\mathbf{r}^{t-1}(u))|$ means the number of pixels in $N(\mathbf{r}^{t-1}(u))$. For the tracking term, we use an estimated scale as a feature:

$$\mu_K = s^{t-1} \tag{3.43}$$

where s^{t-1} is the scale at time $t-1$ as mentioned in (3.6). That is, motion vectors for large objects may be large and the tracking performance using SAD of local patches is likely to be poor. The appearance term has the discriminative power when the color distribution of each label is distinguishable. Therefore, we set the feature to the similarity between histograms of different labels:

$$\mu_A = \frac{1}{\begin{pmatrix} |\mathcal{L}| - 1 \\ 2 \end{pmatrix}} \sum_{l_i, l_j \in \mathcal{L}} \sum_{\mathbf{v}} \min(p^t(\mathbf{v}|l_i), p^t(\mathbf{v}|l_j)) \tag{3.44}$$

where $\mathbf{v}$ is a bin index for $32 \times 32 \times 32$ RGB histograms (this similarity measure is the histogram intersection). Finally, we limit the range of each weight by clipping λ_i to a range $[m_i, M_i]$. This adaptive scheme is compared to a fixed parameter setting on various sequences. As will be shown, this adaptive approach improves the overall tracking performance.

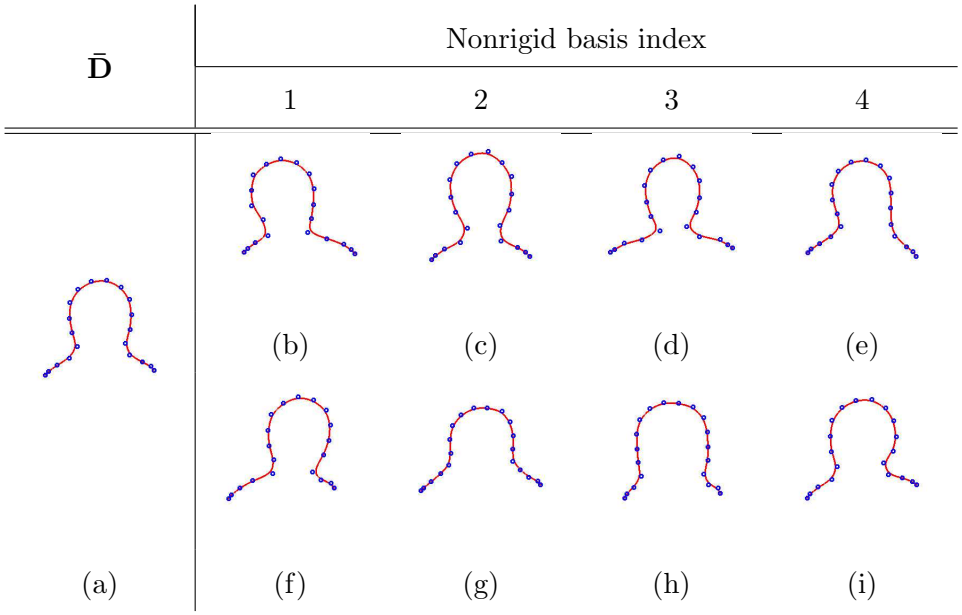


Figure 3.6: Illustration of nonrigid basis vectors: (a) mean shape $\bar{\mathbf{D}}$, (b)-(e) contours corresponding to $\bar{\mathbf{D}} + \text{reshape}(\mathbf{W}(0, \cdots, 0, 2\sigma_i, 0, \cdots, 0)^\top)$, (f)-(i) contours corresponding to $\bar{\mathbf{D}} - \text{reshape}(\mathbf{W}(0, \cdots, 0, 2\sigma_i, 0, \cdots, 0)^\top)$.

3.4 Experimental results

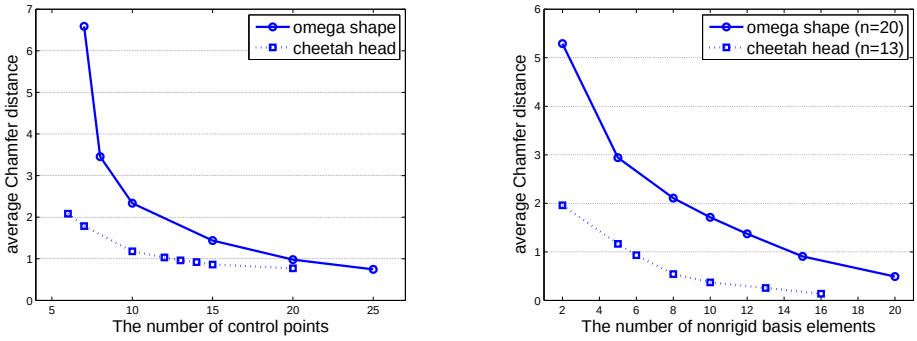
We have conducted extensive experiments on a human omega shape and a cheetah profile shape in the videos available at `http://ispl.snu.ac.kr/hsfra111/opencontour/`, where all the experimental results of our algorithm including failure cases are also available. For the training, we have collected 205 and 134 samples for a human omega shape and a cheetah profile shape respectively, and estimated the parameters in the proposed state space model. In Fig. 3.6, we depict four nonrigid basis vectors by showing contours corresponding to

$$\mathbf{D} = \bar{\mathbf{D}} \pm \text{reshape}(\mathbf{W}(0, \cdots, 0, 2\sigma_i, 0, \cdots, 0)^\top) \tag{3.45}$$

where σ_i means the standard deviation of the i-th element of $\mathbf{q}$ as mentioned in (3.12). Also, we implemented a simple recovery logic. When a posterior probability $p(\boldsymbol{\omega}^t|I^{1:t})$ goes below a certain threshold, we consider that the tracking fails and more particles are scattered with a larger covariance matrix $\mathbf{S}$. We implemented our method with C++ and it runs at 10 frames per second under VGA resolution on a general-purpose PC (AMD Phenom(tm) II x6 1055T 2.8Ghz). The proposed method consists of three blocks: pre-processing (image filtering, contrast computation and pixel-wise classification $J(I^t)$), cost function evaluations for particles, and model update. The cost function evaluation block takes about 70% of the processing time and the construction of the label map $J(\boldsymbol{\omega}^t)$ for $E_A(\cdot)$ is the bottleneck in the cost function evaluation block. It takes about 70% of the block processing time.

3.4.1 Parameter selection

In order to determine the number of control points n and the number of nonrigid basis k discussed in Section 3.1, we compute the Chamfer distance from the ground



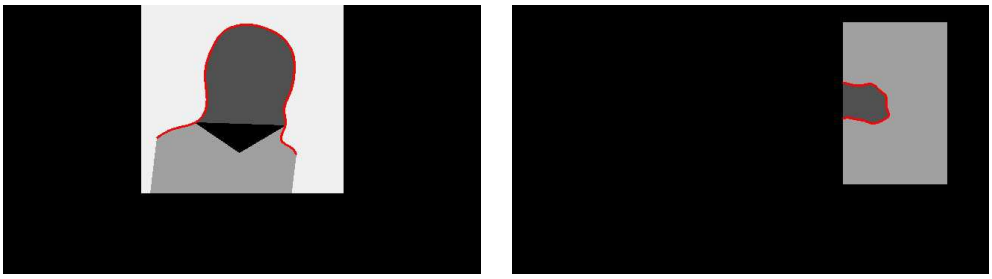
(a) (b)

Figure 3.7: Chamfer distances for varying parameters: (a) Chamfer distances from ground truth to reconstructed contours for several n (the number of control points), (b) Chamfer distances from original B-spline contours to the contours reconstructed with k basis vectors.

truth boundaries to the reconstructed ones [41]:

$$\text{dist}(G, E) \propto \sum_{\mathbf{u} \in G} \min_{\mathbf{v} \in E} \|\mathbf{u} - \mathbf{v}\|_1 \tag{3.46}$$

where G represents the ground truth contour and E stands for the estimated contour. As shown in Fig. 3.7, the increase in n and k results in the decrease of distance, and we set the parameters so that the average distance is less than 1 pixel ($n = 20$ and $k = 15$ for a human omega shape and $n = 13$ and $k = 6$ for a cheetah profile). We also have to select 12 parameters $\{a_i, b_i, m_i, M_i\}$ ($i \in \{C, K, A\}$) for the adaptive weight scheme discussed in Section 3.3.5, where the exhaustive search is too time consuming. Therefore, we adopt a simple but effective method. First, we select the range of weight parameter λ_i, i.e. $\{m_i, M_i\}$, so that the method shows good performance (low Chamfer distance) for two training sequences "girl2" and "soccer1". Then, we find the range of indicating feature μ_i on the training



(a) (b)

Figure 3.8: Examples of $J(\omega^t)$ for (a) a human omega shape and (b) a cheetah profile shape. Black pixels stand for "don't care" regions.

sequences. Finally, we select parameters $\{a_i, b_i\}$ so that the linear relation (3.41) yields one-to-one correspondences between two intervals (ranges): $a_C = -0.297$, $b_C = 0.695$, $a_K = -0.0122$, $b_K = 5.365$, $a_A = -505.05$, $b_A = 90.51$, $\lambda_C \in [5, 25]$, $\lambda_K \in [0.5, 5]$, and $\lambda_A \in [40, 80]$. Other parameters were set empirically: $\mathbf{S} = \mathrm{diag}(0.05^2, 0.05^2, 0.02^2, 0.02^2)$, $\lambda_R = 10$, $\gamma = 0.02$, and $\beta = 0.01$. In addition, 800 particles were used in the particle filtering.

3.4.2 Label map construction

For the evaluation of the measurement function, we need a label map $J(\boldsymbol{\omega}^t)$. However, this is an application-dependent task, and we manually build a (template) label map and transform the map according to $\boldsymbol{\omega}^t$ as shown in Figs. 3.8-(a) and (b). To be precise, we used four labels for a human omega shape: $\mathcal{L} = \{$"head", "body", "background", "don't care"$\}$ and three labels for a cheetah profile shape: $\mathcal{L} = \{$"head", "background", "don't care"$\}$. This requirement may seem to ask much user interaction. However, note that this step is needed only once for each target category. That is, we have a standard template $J(\boldsymbol{\omega}_0)$ which is defined using

the contour $\mathbf{r}_0(\cdot)$ and some knot points, then $J(\boldsymbol{\omega^t})$ can be obtained by transforming $J(\boldsymbol{\omega}_0)$ with a transformation that can be estimated from ω_0 and ω^t. Specifically, we can draw the contour $\mathbf{r}^t(\cdot)$ and knot points of $\mathbf{r}^t(\cdot)$ from ω^t, and the label map $J(\boldsymbol{\omega^t})$ can be built from $\mathbf{r}^t(\cdot)$ and knot points for any ω^t. This method allows us to exploit the characteristic of target objects. For example, colors of faces and upper body are usually different, and we can exploit this feature in the tracking by assigning different labels to them as shown in Fig. 3.8-(a) (detailed description can be found in the project page).

3.4.3 Comparison with existing contour-based methods

The proposed method is compared with the conventional algorithms that provide boundary information [15–17]. In [15], the open contour is parameterized with B-splines and its deformation is modeled with a linear combination of basis vectors. For a fair comparison, we trained their model with the same training set as ours. The methods in [16, 17] are video segmentation methods, which yield closed contours.

For the evaluation, we compute the Chamfer distance from the ground truth to the extracted contours (segmentation boundaries for video segmentation algorithms). We used eight sequences, whose representative frames are shown in Fig. 3.9. We tried to collect a variety of sequences: the "news" sequence is a static sequence; "girl2," "man," "ys," "cheetah1," and "cheetah3" sequences show a large amount of nonrigid deformations; "thr" and "ys" sequences show large scale and orientation changes; "girl2," "man," and "ys" sequences have complex backgrounds; and occlusions frequently appear in "cheetah2" and "cheetah3" sequences. The results are summarized in Tables 3.1 and 3.2 where "fixed/adaptive" indicates the weight selection method discussed in Section 3.3.5. It is clear from the results

(a) girl2

(b) man

(c) news

(d) ys

(e) thr

(f) cheetah1

(g) cheetah2

(h) cheetah3

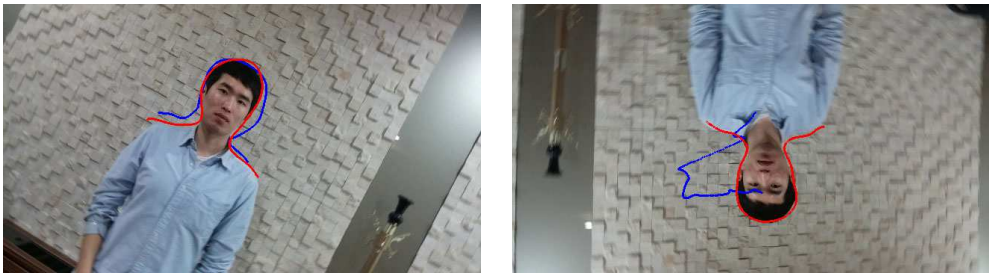
Figure 3.9: Representative frames of some test sequences.

Table 3.1: The average Chamfer distance for human omega shape sequences. The best results are in boldface. We use a symbol * to represent sequences used in adaptive weight learning.

Sequences	Condensation [15]	Snapcut [16]	HT [17]	Proposed (fixed/adaptive)
girl2	75.385	**2.614**	17.798	3.978 / 3.260(*)
man	20.302	25.218	11.677	**4.389** / 5.256
news	2.826	1.481	2.973	1.632 / **1.386**
ys	39.459	13.246	15.255	4.934 / **3.788**
thr	44.431	7.352	13.048	3.393 / **2.704**
Average	36.481	9.982	12.150	3.665 / **3.279**

that the proposed method shows improved tracking performance compared with the B-spline based method [15]. We believe this is because (a) the proposed method exploits colors as well as contrast which makes it more robust to background clutters and (b) the proposed method can handle rigid motions such as scale changes and in-plane rotations in a geometrically meaningful manner (e.g., 360° in-plane rotation on the "thr" sequence). Also, we can see that the adaptive weight scheme usually yields better results than the fixed weight one, showing that the weight adaptation is an effective way to improve the performance. Some results are shown in Fig. 3.10. Video segmentation cannot impose shape constraints during the tracking and it fails to deal with nonrigid shapes as shown in Fig. 3.11, besides showing large Chamfer distances compared with the proposed method.

(a) girl



(b) thr

Figure 3.10: Blue and red curves represent the results of [15] and the proposed method, respectively.

Table 3.2: The average Chamfer distance for cheetah profile sequences. The best results are in boldface.

Sequences	Condensation [15]	Snapcut [16]	HT [17]	Proposed (fixed/adaptive)
cheetah1	11.914	**1.125**	2.977	1.687 / 1.336
cheetah2	7.092	64.647	3.963	1.811 / **1.751**
cheetah3	96.352	13.288	69.659	9.351 / **8.254**
Average	38.452	26.353	25.533	4.283 / **3.78**

(a) girl2 (b) man

(c) news (d) ys

Figure 3.11: Tracking results on human sequences. The first and second rows show the results of [16] and [17]. The bottom row shows the result of the proposed method.

3.4.4 Comparison with bounding box based methods

The proposed method provides boundary information, and it has an advantage over conventional bounding box tracking methods as discussed in the introduction. However, the proposed method also shows good tracking performance in terms of location estimation for some targets. For the objective evaluation, we use the Jaccard index (JI score):

$$\mathrm{JI}(A_{gt}, A_{est}) = \frac{|A_{gt} \cap A_{est}|}{|A_{gt} \cup A_{est}|} \tag{3.47}$$

where A_{gt} and A_{est} stand for the ground truth box and the estimated box respectively. When the index in (3.47) is greater than a given threshold, we consider the estimation is correct (0.5 is used for the threshold). Since we generate A_{est} from the estimated contours in order to compare the performance with conventional methods, the generation of A_{est} depends on target objects and initial tracking boxes.

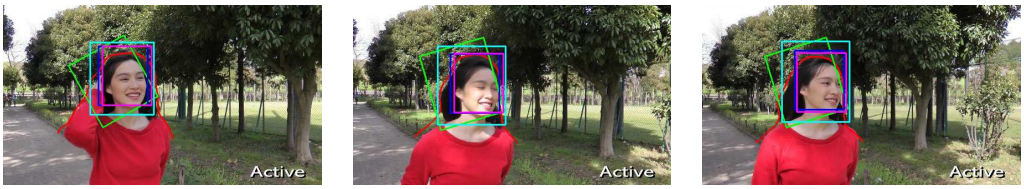
In the experiments, we used 12 videos (4 sequences from [6] and 8 sequences used in the previous section) and 4 conventional methods were evaluated [2–5]. Experimental results obtained with a human omega shape are summarized in Table 3.3 and some tracking results are shown in Fig. 3.12. In the experiments, the initial boxes of conventional methods only contain face regions, and therefore A_{est} of the proposed method is determined as a quadrangle covering face regions, which is given by the knot points of $\mathbf{r}^t(u)$ (refer to our project page for details). Compared with other sequences, the proposed method does not show good performance for the "david" sequence. The sequence is grayscale and shows a large amount of illumination changes so that the boundaries are not clearly shown. Therefore, our measurement terms $E_C(\cdot)$ and $E_A(\cdot)$ has difficulties to localize targets. In order to handle a large amount of translations (fast motions), the rigid motion parameter

Table 3.3: Comparison with conventional box tracking methods on human omega shape sequences (The ratio of successfully tracked frames). We use a symbol * to represent sequences used in adaptive weight learning.

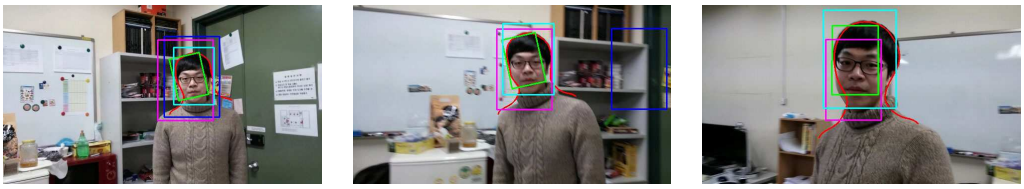
Sequences	IVT [2]	ELK [3]	CSK [4]	FCT [5]	Proposed (fixed/adaptive)
david	0.215	0.624	0.559	**1**	0.526 / 0.656
girl	0.158	**0.941**	0.495	0.307	0.891 / 0.921
faceocc	0.644	0.864	**1**	0.559	**1** / **1**
faceocc2	0.503	0.723	**1**	0.810	0.712 / 0.755
girl2	0.9	**1**	**1**	**1**	**1** / **1**(*)
man	**1**	**1**	**1**	**1**	**1** / **1**
news	**1**	**1**	**1**	**1**	**1** / **1**
ys	0.877	0.877	0.667	0.825	**1** / **1**
thr	**1**	**1**	**1**	0.817	0.983 / 0.983
Average	0.700	0.892	0.858	0.813	0.901 / **0.924**

Table 3.4: Comparison with conventional box tracking methods on cheetah sequences (The ratio of successfully tracked frames).

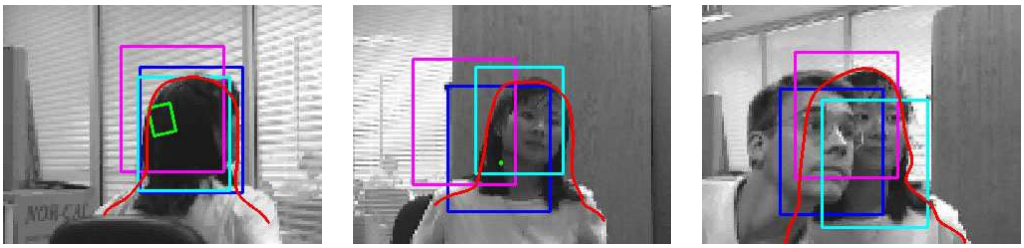
Sequences	IVT [2]	ELK [3]	CSK [4]	FCT [5]	Proposed (fixed/adaptive)
cheetah1	**1**	**1**	**1**	**1**	**1** / **1**
cheetah2	**1**	**1**	**1**	0.889	**1** / **1**
cheetah3	0.455	0.5	0.409	0.318	0.591 / **0.727**
Average	0.818	0.833	0.803	0.736	0.864 / **0.909**



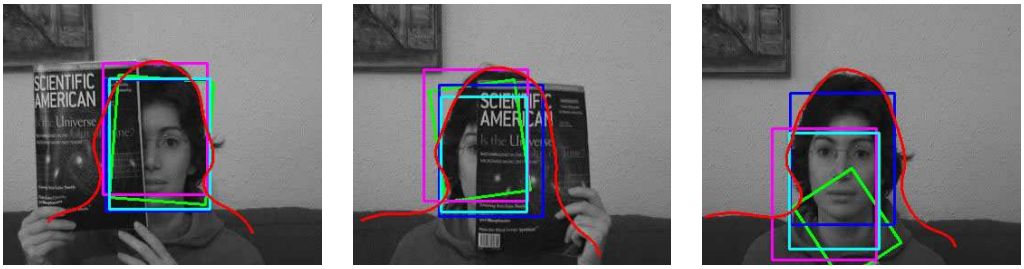
(a) girl2



(b) ys



(c) girl



(d) faceocc

Figure 3.12: Rectangles are the results of box tracking methods with initial boxes of face regions. Green, cyan, blue, magenta, and red denote the results of IVT [2], ELK [3], CSK [4], FCT [5], and the proposed method respectively (best viewed in color on a computer screen).

(a) cheetah1

(b) cheetah2

(c) cheetah3

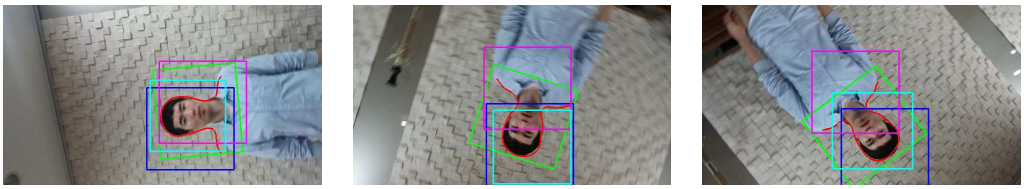
Figure 3.13: Rectangles are the results of box tracking methods and curves are the results of contour tracking methods. (a) Our shape tracking results on the "cheetah1" sequence, (b) our shape tracking results on the "cheetah2" sequence, (c) experimental results on the "cheetah3" sequence: green, cyan, blue, magenta, dark blue, dark green, yellow, white, black, and red represent IVT [2], ELK [3], CSK [4], FCT [5], SPT [9], DGT [10], BHMC [12], Video snapcut [16], HT [17], and proposed method respectively (best viewed in color on a computer screen).

Table 3.5: Comparison with conventional box tracking methods on human omega shape sequences using large initial boxes containing face and shoulder (The ratio of successfully tracked frames). We use a symbol * to represent sequences used in adaptive weight learning.

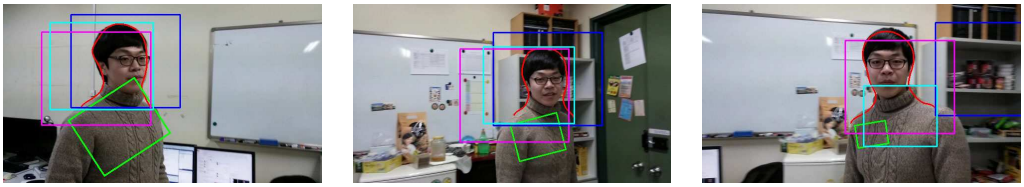
Sequences	IVT [2]	ELK [3]	CSK [4]	FCT [5]	Proposed (fixed/adaptive)
girl2	**1**	0.6	**1**	**1**	**1** / **1**(*)
man	**1**	-	**1**	**1**	**1** /**1**
news	**1**	**1**	**1**	**1**	**1** / **1**
ys	0.246	0.719	0.596	0.772	**1** / **1**
thr	**1**	0.683	0.737	0.917	0.983 / 0.983
average	0.849	0.751	0.867	0.896	**0.997** / **0.997**

of "cheetah2" and "cheetah3" sequences is set to $\mathbf{S} = \mathrm{diag}(0.05^2, 0.05^2, 0.2^2, 0.2^2)$. Evaluation results obtained with cheetah sequences are summarized in Table 3.4. For cheetah sequences, A_{est} of the proposed method is given by the smallest bounding box including the contour $\mathbf{r}^t(u)$. The results show that the proposed method compares favorably with the conventional methods for the human omega shape and the cheetah profile. There are severe occlusions in "girl," "faceocc," "faceocc2," and "cheetah3" sequences, and the results also show that our method successfully handles occlusions. The tracking results of the cheetah shape are shown in Fig. 3.13, and the videos are available at the above stated site.

Finally, we evaluate the tracking performance using initial boxes containing face and shoulders for human sequences. That is, human pose estimation is sometimes



(a) thr (face and shoulder)



(b) ys (face and shoulder)

Figure 3.14: Rectangles are the results of box tracking methods with large initial boxes containing faces and shoulders. Green, cyan, blue, magenta, and red denote the results of IVT [2], ELK [3], CSK [4], FCT [5], and the proposed method respectively (best viewed in color on a computer screen).

necessary and we need to consider larger regions in this case. The results are summarized in Table 3.5 where A_{est} of the proposed method is constructed with a similar manner to the cheetah case. As shown, conventional methods show performance degradation, because larger bounding boxes contain more background pixels. Therefore, the proposed method becomes more beneficial when users want to track larger areas (e.g., for the estimation of poses). Some experimental results for this case are shown in Fig. 3.14.

3.4.5 Comparison with tracking methods for nonrigid objects

Since the proposed method is developed to handle nonrigid motions, we compared the performance with conventional nonrigid object tracking methods [9, 10, 17] on challenging sequences (sequences and result videos are available at our project page). As shown in Fig. 3.15, the target object (human omega shape) experiences nonrigid deformations, moves and scales abruptly, and sometimes occluded by other people. For the conventional methods, we initialize the tracking box so that it contains face and shoulder regions. The performance is also evaluated with Jaccard index (0.5 is used as the threshold). Estimated boxes of the proposed method and the method in [17] are considered the smallest bounding boxes containing contour results. The comparisons results are shown in Table 3.6 and Fig. 3.15. As shown, the proposed method shows superior performance compared with [9, 10, 17]. Although the proposed method lost the target objects in "dance3" and "dance4" sequences due to occlusions and camouflage, they were recovered soon by our failure recovery logic as shown in the last row of Fig. 3.15 (videos showing the results are also available in the project page).

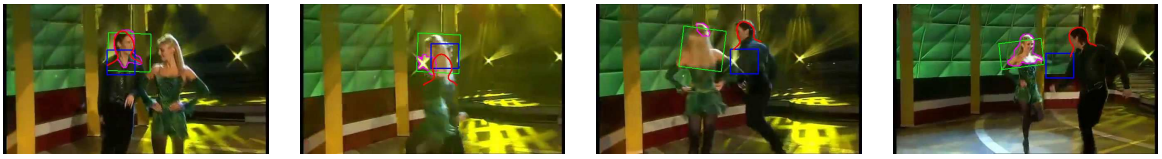
Table 3.6: Comparison with conventional box tracking methods for nonrigid objects on human omega shape sequences using large initial boxes containing face and shoulder (The ratio of successfully tracked frames). We use a symbol * to represent sequences used in adaptive weight learning.

Sequences	HT [17]	SPT [9]	DGT [10]	Proposed (fixed/adaptive)
soccer1	0.143	0.571	**1**	**1** / **1**(*)
soccer2	**0.5**	0.2	0.2	0.3 / **0.5**
soccer3	0.222	0.278	0.111	0.944 / **1**
soccer4	0.105	0.211	0.105	0.368 / **0.474**
soccer5	0.667	0.222	0.111	**1** / **1**
soccer6	0.067	0.333	0.067	**1** / **1**
soccer7	0.1	0.6	0.1	**0.9** / **0.9**
dance1	0.74	0.18	0.12	0.92 / **0.98**
dance2	0.467	0.048	0.01	**1** / 0.99
dance3	0.12	0.12	0.16	0.92 / **0.96**
dance4	**0.75**	0.208	0.25	0.708 / 0.625
dance5	0.12	0.08	0.04	**0.4** / **0.4**
average	0.333	0.254	0.19	0.788 / **0.819**

(a) soccer3

(b) soccer6

(c) dance2



(d) dance3

Figure 3.15: The results of conventional tracking methods for nonrigid objects and the proposed method are shown: (a) soccer3, (b) soccer6, (c) dance2 and (d) dance3 sequences. Magenta, green, blue and red results indicate HT [17], SPT [9], DGT [10] and the proposed method respectively.

Chapter 4

Target-dependent closed contour tracking

4.1 The proposed model

In order to design target-based closed contour tracking, the same state space and the framework of an open contour case are adopted. The state vector $\boldsymbol{\omega}^t = \{\mathbf{X}^t, \mathbf{q}^t\}$ consists of rigid transformation matrix $\mathbf{X}^t$ and the coefficient vector $\mathbf{q}^t$ for non-rigid deformation, and they are definitely separated and handled independently. The state vector is updated based on the recursive Bayesian estimation [15] in (1.1).

4.1.1 Active contour model

Similar to the open contour tracking, closed contours are also represented by B-spline model that have n basis functions $\mathbf{B}(u)$ and n control points $\mathbf{D}$ in (3.2) and the contour $\mathbf{r}(u)$ is defined on $u \in [0, L]$. B-spline models for the closed contour are different from the ones for the open contour that periodic cubic spline functions are

chosen to represent the closed contour. Therefore, $L = n$ and each basis function is built by translating the original basis function $B_1(u)$:

$$B_1(u) = \begin{cases} \frac{1}{6}u^3 & \text{when} \quad 0 \le u < 1 \\ -\frac{1}{2}(u-1)^3 + \frac{1}{2}(u-1)^2 + \frac{1}{2}(u-1) + \frac{1}{6} & \text{when} \quad 1 \le u < 2 \\ \frac{1}{2}(u-2)^3 - (u-2)^2 + \frac{2}{3} & \text{when} \quad 2 \le u < 3 \\ \frac{1}{6}(1-(u-3))^3 & \text{when} \quad 3 \le u < 4 \\ 0 & \text{when} \quad 4 \le u < L \end{cases} \tag{4.1}$$

where $B_1(u)$ is a periodic function with period L and other basis functions are defined as

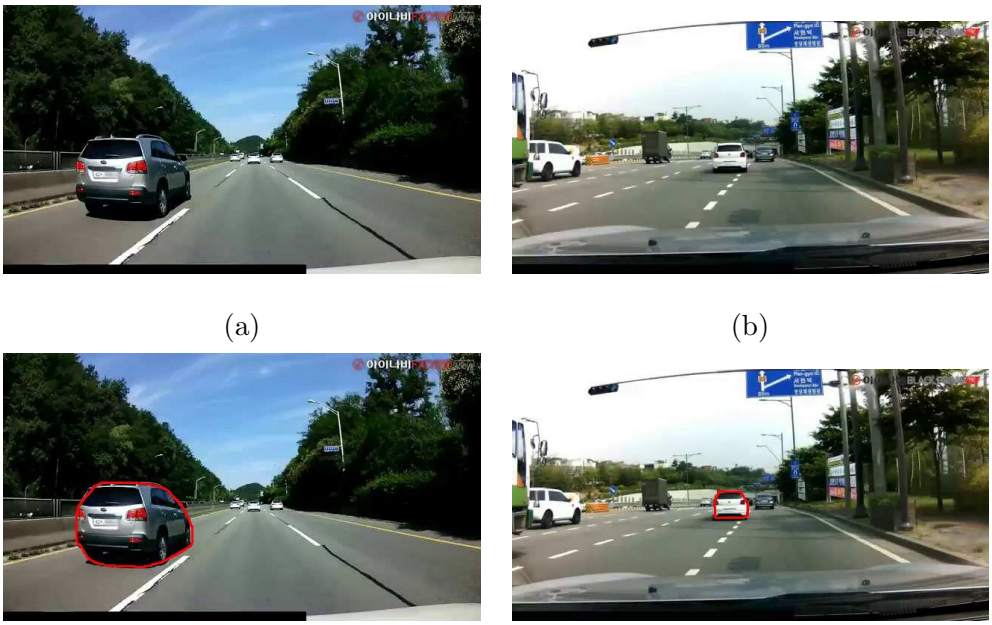
$$B_i(u) = B_1(u - i + 1) \quad \text{for} \quad 1 \le i \le n. \tag{4.2}$$

Fig. 4.1 shows examples of objects, their corresponding closed contour $\mathbf{r}(\cdot)$ and n control points $\mathbf{D}$.

4.1.2 Contour dynamics

The closed contour deforms its shape rigidly or non-rigidly, and the shape generating model is proposed to estimate the shape variations in (3.5). In this model, rigid and non-rigid motions of closed contour are absolutely separated from each other where rigid motion is conducted with a geometrically well defined model and non-rigid deformation is implemented by adding non-rigid motion bases to the mean shape. The mean shape $\bar{\mathbf{D}}$ and non-rigid motion basis $\mathbf{W}$ can be obtained by a training step which is already mentioned in Section 3.2 and Algorithm 1.

Also, the proposed method adopts the stochastic dynamic model where rigid and non-rigid states are independent in as (3.7). The dynamic model for the rigid state $p(\mathbf{X}^t|\mathbf{X}^{t-1})$ is designed with an approximated normal distribution on $Aff(2)$



(a) (b)

(c) (d)

(e) (f)

Figure 4.1: First row: input images. Second row: ground truth contours that are drawn manually. Third row: estimated contours $\mathbf{r}(u)$ (red curve) and their control points $\mathbf{D}$ (blue circles).

and the non-rigid state dynamic model $p(\mathbf{q}^t \mid \mathbf{q}^{t-1})$ consists of Gaussian and Laplace distributions to prevent overfitting.

4.2 Measurement

Measurement is designed considering object appearance, image contrast and temporal coherency. Therefore, the measurement model $p(I^t|\boldsymbol{\omega}^t)$ is evaluated as (3.34) where E_C emphasizes contrast, E_K means local track energy and E_A represents object appearance. In the closed contour tracking, E_C and E_A terms are same to the ones in (3.35) and (3.38) in the open contour tracking method, however the local track term E_K is developed in a different way.

4.2.1 Local track term

The target object may look different during the tracking due to occlusion, illumination change, changing appearances and view point. Therefore, learning the object with only whole appearance is a very risky way and it will miss the target when appearances are changed. If some parts of the object are considered and trained individually, then it would be more robust to learn object appearance.

According to above idea, the proposed method learns partial object appearances that are modeled by a set of local patches and its relative positions:

$$P^t = \{(f^t_{p,i}, f^t_{k,i})\}_{i=1}^{N_f} \tag{4.3}$$

where $s^t * f^t_{p,i}$ means the relative position to the center of control points ($\frac{1}{n}\mathbf{D}^t\mathbf{1}_{n\times 1}$), s^t is the scale at time t as mentioned in (3.6), $f^t_{k,i}$ presents the local square patch with a fixed size l_k and N_f are the number of patches. Then, given $\mathbf{D}^t$, the image patch whose center point locates at $\frac{1}{n}\mathbf{D}^t\mathbf{1}_{n\times 1} + s^t * f_{p,i}$ is picked with a size $\frac{s^t}{s_k} l_k$

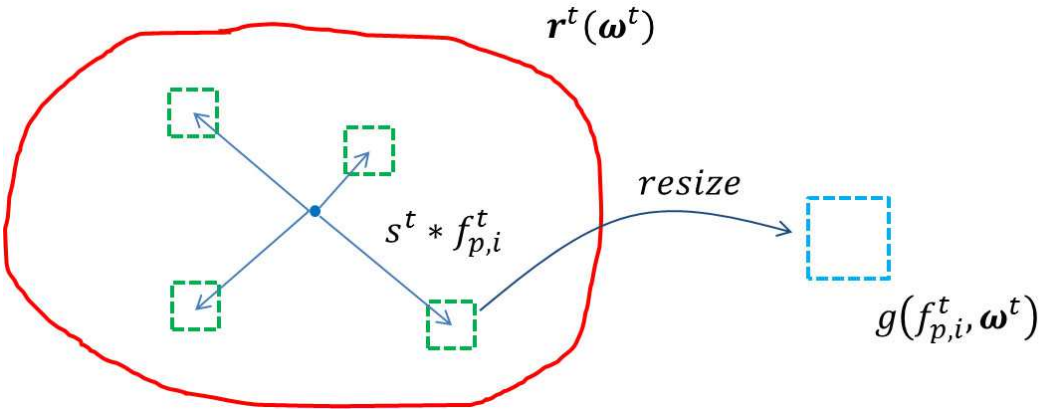


Figure 4.2: This illustration represents notations for the local track term. Red contour $\mathbf{r}^t(\cdot)$ stands for the object shape, blue dot is the center of control points $\frac{1}{n}\mathbf{D}^t\mathbf{1}_{n\times 1}$ and green squares presents local patches that are located at $s^t * f_{p,i}^t$ from the center of control points. Each local patch is extracted and resized to $g(f_{p,i}^t, \boldsymbol{\omega}^t)$ (cyan square) that corresponds to $f_{k,i}^t$.

and resized to the size l_k where s_k stand for the scale which corresponds to the size l_k. This resized patch is denoted as $g(f_{p,i}^t, \boldsymbol{\omega}^t)$. Finally, the local track energy is calculated using SAD criterion between corresponding patches:

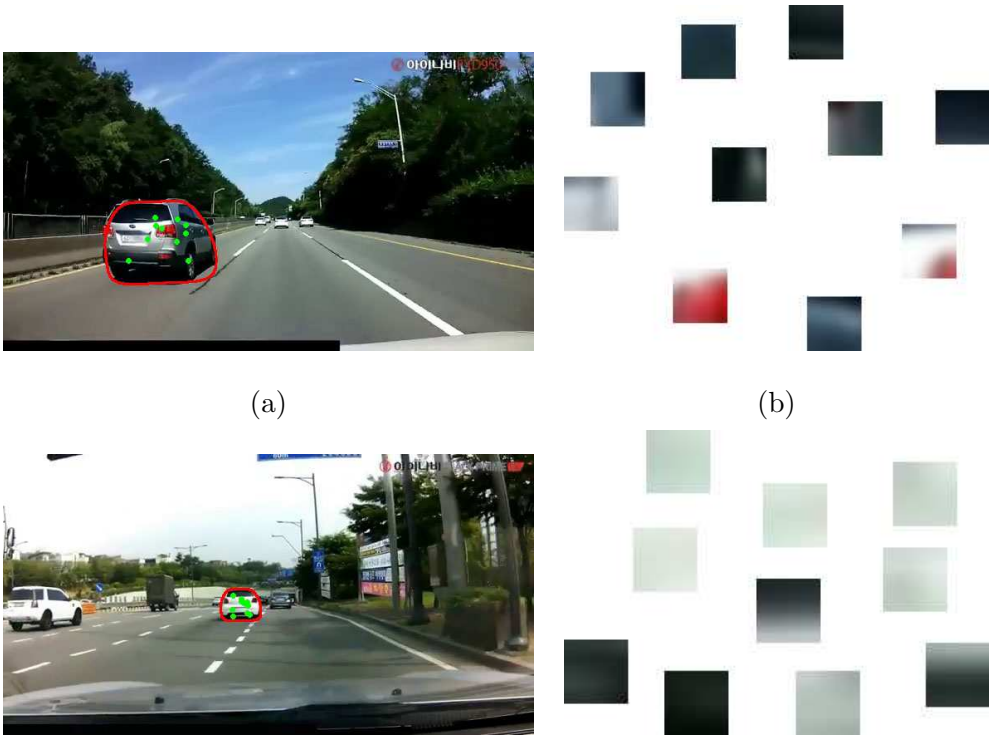
$$E_K(\boldsymbol{\omega}^t) = \frac{1}{N_f}\sum_{i=1}^{N_f}\frac{1}{l_k^2}\|f_{k,i}^t - g(f_{p,i}^t, \boldsymbol{\omega}^t)\|_1 \tag{4.4}$$

where Fig. 4.2 can be helpful to understand this local track term.

At the end of the frame, if $g(f_{p,i}^t, \boldsymbol{\omega}^t)$ is similiar to $f_{k,i}^t$, then the local appearances model P^t is updated with the following equation:

$$f_{k,i}^{t+1} = (1-\alpha_k)f_{k,i}^t + \alpha_k g(f_{p,i}^t, \boldsymbol{\omega}^t) \tag{4.5}$$

where $i \in [1, N_f]$ is the integer. However, the element $(f_{p,i}^t, f_{k,i}^t)$ is replaced to the new one under following conditions:



(a) (b)

(c) (d)

Figure 4.3: The example of local appearance model P^t is represented. (a), (c) Green dots mean the locations of local patches and (b), (d) some local patches $f^t_{k,i}$ are provided.

- the distance between $f^t_{k,i}$ and $g(f^t_{p,i}, \boldsymbol{\omega}^t)$ is far from the threshold

- each element of P^t can be substituted with a certain probability.

When the new local appearance model is drawn, $f^t_{p,i}$ is chosen at random satisfying that the drawn patch should be located in the closed contour. Examples of P^t are shown in Fig. 4.3.

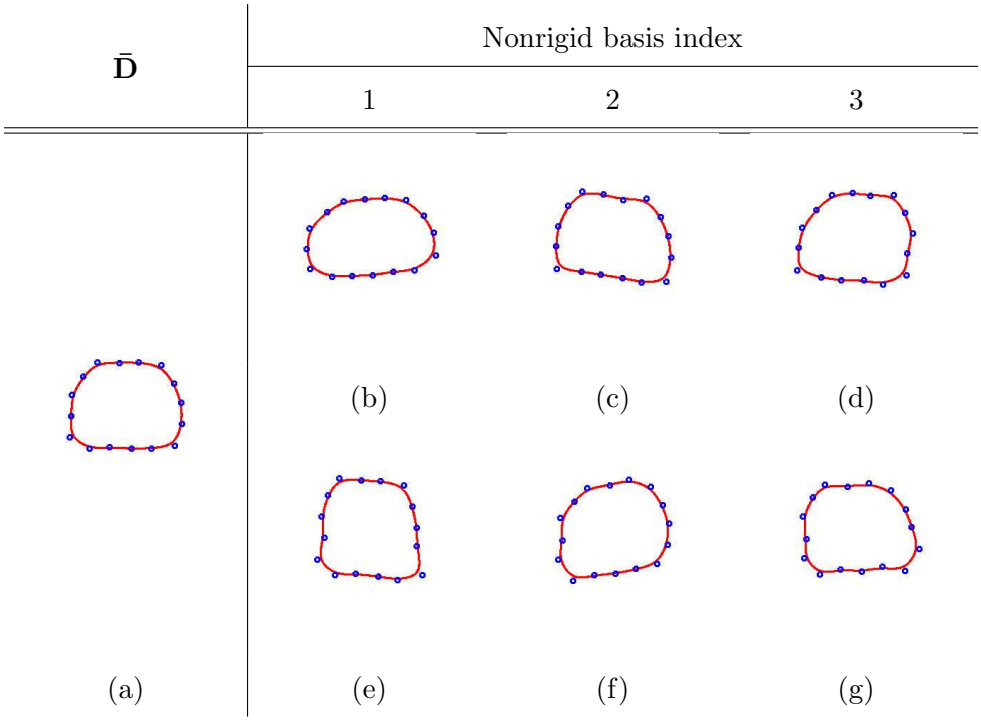
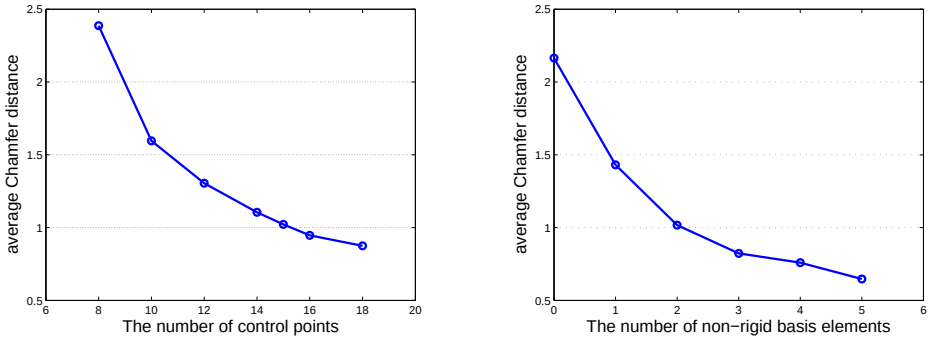


Figure 4.4: Illustration of nonrigid basis vectors for cars in black-box video: (a) mean shape $\bar{\mathbf{D}}$, (b)-(d) contours corresponding to $\bar{\mathbf{D}}$ + reshape($\mathbf{W}(0, \cdots, 0, 2\sigma_i, 0, \cdots, 0)^\top$), (e)-(g) contours corresponding to $\bar{\mathbf{D}}$ − reshape($\mathbf{W}(0, \cdots, 0, 2\sigma_i, 0, \cdots, 0)^\top$).

4.3 Experimental results

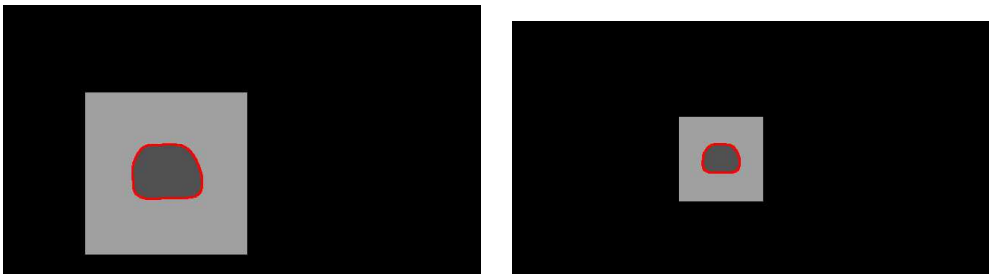
In order to evaluate the proposed method, we select the target object as cars captured by blackboxes, whose boundary shape can be deformed non-rigidly depending on the view point of blackboxes. 170 closed contours of cars are drawn manually and used for the training process to find the mean shape $\bar{\mathbf{D}}$ and the non-rigid shape basis $\mathbf{W}$. Shape examples and their control points that are constructed using the mean shape and the non-rigid basis are demonstrated in Fig. 4.4.



(a) (b)

Figure 4.5: Chamfer distances for varying parameters: (a) Chamfer distances from ground truth to reconstructed contours for several n (the number of control points), (b) Chamfer distances from original B-spline contours to the contours reconstructed with k basis vectors.

Similar to Section 3.4.1, the number of control points n and the number of non-rigid basis k are determined by measuring the average Chamfer distances which are less than 1 pixel as shown in Fig. 4.5. First, the contour which is drawn with n control points is compared to the ground truth contour, and $n = 16$ is chosen. Then, k is set to 3 by evaluating the average Chamfer distance from the fitted contour with n control points to the estimated contour with k basis vectors. The parameters for local track term are set to $N_f = 10$ and $\alpha_k = 0.04$, and three weight parameters of measurements are determined automatically with our adaptive scheme in Section 3.3.5. Other parameters are same to the open contour case as mentioned in Section 3.4.1 except the number of particles that 400 particles are scattered to increase the operation speed. The proposed method is implemented with C++ and runs at 10 frames per second under VGA resolution images on a general PC (AMD Phenom(tm)



(a) (b)

Figure 4.6: Given the closed contour, the label map $J(\boldsymbol{\omega}^t)$ is constructed. The intensity in $J(\boldsymbol{\omega}^t)$ means the label and black pixels present "don't care" region.

II x6 1055T 2.8Ghz). The composition of closed contour tracking is similar to the algorithm for open contour tracking and the cost evaluation block also takes about 70% of the processing time.

4.3.1 Label map construction

In the closed contour tracking, the label map $J(\boldsymbol{\omega}^t)$ can be constructed with a simple criterion. Given ω^t, we can distinguish the inner region (real object part) from the outer area (background) easily. Therefore, we just use three labels: $\mathcal{L} = \{$"object", "background", "don't care"$\}$ that "object" is set to be inside of the contour $\mathbf{r}^t(\cdot)$, "background" is assigned to outside of $\mathbf{r}^t(\cdot)$ and "don't care" is set to the outer area of ROI. Fig. 4.6 shows examples of $J(\boldsymbol{\omega}^t)$.

4.3.2 Comparison to conventional tracking methods

The proposed method is tested on 9 blackbox sequences, where 5 of them are captured in the daytime and others are recorded at night as shown Fig. 4.7. Test sequences are collected by considering various environments, for example, the scale of

(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 4.7: Representative frames of test sequences are presented.

the car gets smaller in "day1", "day3", "night1", "night3" and "night4" sequences, and larger in "day5" sequence, illumination and contrast are changed in "day2", "night2" and "night4" sequences, there exists occlusions in "day4" sequence, and the car experiences non-rigid deformations in the most sequences. We compare the proposed method to conventional box tracking algorithms [2, 4] and contour-based methods [16, 17] that are described in Chapter 3. In order to evaluate the performance, the Jaccard index which is introduced in (3.47) is selected and the tracking is rated as success when the Jacarrd index is larger than the threshold (we choose 0.5 as the threshold). Table 4.1 and 4.2 provide the ratio of successfully tracked frames and the average Jaccard index over all frames. Note that [4] usually has trouble to track the object when the object size is changing because it does not consider the scale variation. Also, [16] brings wrong estimates of the object region when the contrast between object and background is low. Since it is difficult to distinguish the object from the background at night, most methods yield poor results in night sequences. The proposed method shows the best performance among compared methods as shown Table 4.1 and 4.2. Especially, the proposed method presents more remarkable values in Table 4.2 that it achieves to track the boundary of the object robustly and accurately as well as the location with our shape model and measurements. Illustration examples for combining results of compared methods are provided in Fig. 4.8.

Table 4.1: Comparison with conventional tracking methods for the ratio of successfully tracked frames.

Sequences	IVT [2]	CSK [4]	Snapcut [16]	HT [17]	Proposed
day1	**1**	0	**1**	0.417	**1**
day2	**1**	0.818	0.5	0.864	0.955
day3	**1**	0.120	0	0.520	**1**
day4	**0.978**	0.889	0.222	0.933	0.956
day5	0.667	0.667	0	0.333	**1**
night1	0.462	0.231	**1**	0.308	**1**
night2	0.688	0.125	0.688	0.5	**1**
night3	**1**	0.133	0.533	0.400	0.867
night4	0.765	0.294	0.588	0.294	**0.941**
average	0.840	0.364	0.503	0.508	**0.969**

Table 4.2: Comparison with conventional tracking methods for the average JI score.

Sequences	IVT [2]	CSK [4]	Snapcut [16]	HT [17]	Proposed
day1	0.656	0.147	0.844	0.425	**0.870**
day2	0.793	0.569	0.367	0.642	**0.798**
day3	0.605	0.260	0.249	0.552	**0.844**
day4	0.761	0.655	0.423	0.672	**0.807**
day5	0.504	0.546	0.262	0.370	**0.812**
night1	0.483	0.251	0.833	0.340	**0.882**
night2	0.525	0.358	0.588	0.405	**0.770**
night3	0.532	0.326	0.531	0.417	**0.651**
night4	0.572	0.357	0.563	0.418	**0.738**
average	0.603	0.385	0.518	0.471	**0.797**

(a) day1

(b) day5

(c) night1

(d) night2

Figure 4.8: Tracking results of conventional tracking methods and the proposed method are represented in the sequence of (a) “day1”, (b) “day5”, (c) “night1” and (d) “night2”. Green, blue, cyan, yellow and red curves correspond to IVT [2], CSK [4], Snapcut [16], HT [17] and the proposed method.

4.4 Special case : document capture

When the shape of target object is well known in advance and easy to model with some variables, the training process for shape variation is unnecessary and we can use simple contour model. The document is a good example of a well known closed contour, which is shown as a quadrangle and can be modeled with only 4 corner points. Therefore, we provide an application for document capture that is an interesting topic in computer vision field because there exists innumerable camera captured document images along with the digital device development such as smartphone and tablet, and automatic system is required for handling them easily. In the proposed method, document capture algorithm is designed by a tracking-by-detection manner with a known shape and some assumptions, and this study can be utilized as a pre-processing step for other document image processing.

In this dissertation, we use unified notations except only this section. Note that notations in this section are different from other sections. It is no matter to regard this section as an independent one.

4.4.1 Document model

Since the document has a quadrilateral shape generally, the deformation of closed contour is limited on the quadrilateral space. Therefore, at time t, the state vector $\omega^t = \{\mathbf{p}_{bl}^t, \mathbf{p}_{tl}^t, \mathbf{p}_{tr}^t, \mathbf{p}_{br}^t\}$ has 4 points which are bottom-left, top-left, top-right and bottom-right points and the document region is defined as $\Omega^t = \square\mathbf{p}_{bl}^t\mathbf{p}_{tl}^t\mathbf{p}_{tr}^t\mathbf{p}_{br}^t$. Also, left, right, top, bottom edges are denoted as $Q^t = \{q_L^t, q_R^t, q_T^t, q_B^t\}$. Fig. 4.9 shows an example of document model.

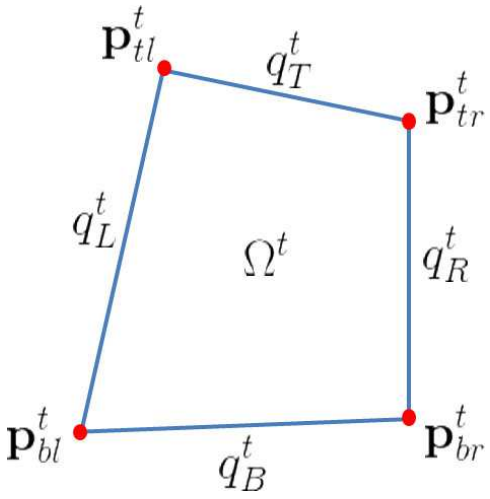


Figure 4.9: Document is modeled by a quadrilateral with four red points and four blue edges.

4.4.2 Document proposals

In order to find the exact document region, many document proposals are generated and tested by their costs. First, noise and textures are removed out by applying morphological close operations

$$J^t = F(I^t) \tag{4.6}$$

where I^t is a given image, J^t is a filtered image, $F(\cdot)$ means the filtering operator and superscript t denotes the time. Because it is generally assumed that the document is a white paper, texts and pictures on the document and other backgrounds outside the document have darker colors than the white paper. Therefore, unnecessary textures can be removed out by the filtering process. Then, line segments that are estimated to document boundaries are extracted by the line segment detector [42] which is applied to a channel that shows the highest contrast among three (red, green, blue) color channels. The contrast of each channel is simply calculated by

the sum of gradient magnitude at each pixel as follows:

$$\text{channel contrast}_i = \sum_{\mathbf{x} \in I^t} \|\nabla J_i^t(\mathbf{x})\| \tag{4.7}$$

where J_i^t means i color channel. The set of extracted line segments $\mathcal{L}^t$ is divided into two sets (horizontal and vertical line segments $\mathcal{L}_h^t$, $\mathcal{L}_v^t$) by computing their orientations. Finally, four line segments (two are included in $\mathcal{L}_h^t$ and the others are from $\mathcal{L}_v^t$) are selected to make lines and four intersection points of horizontal and vertical lines are set to a proposal state vector. Document proposals are created by all possible configurations of line segments, however, inappropriate proposals are excluded considering the positions of line segments and intersection points.

4.4.3 Measurement

For the robust detection and tracking of the document boundaries, a cost function that consists of three measurement terms is proposed. Using the proposed cost functions, the optimization problem is formulated in order to adopt the best proposal as follows:

$$\tilde{\omega}^t = \arg\min_{\omega^t} \lambda_D E_D(\omega^t; \theta^t) + \lambda_C E_C(\omega^t; \mathcal{L}) + \lambda_T E_T(\omega^t; \hat{\omega}^{t-1}) \tag{4.8}$$

where the data term E_D considers color similarity over document and background regions given their color distributions θ^t, E_C prefers boundaries showing high contrast and E_T implies temporal coherency between consecutive frames with previous estimated state vector $\hat{\omega}^{t-1}$. For the optimization, the proposed cost function is evaluated for all proposals and the one showing the lowest value is considered as the solution.

Data term

For the data term, document and background color distributions θ^t are built among each color channel respectively. To be precise, θ^t consists of six distributions

$$\theta^t = \{p_i^t(\cdot|l)\} \quad \text{for} \quad i \in \{r, g, b\}, \quad l \in \{\mathcal{F}, \mathcal{B}\} \tag{4.9}$$

where $\mathcal{F}$ and $\mathcal{B}$ present document and background regions. The data term E_D is defined as

$$E_D(\omega^t; \theta^t) \propto -\sum_{\mathbf{x} \in I^t} \ln p^t(I(\mathbf{x})) \tag{4.10}$$

$$p^t(I(\mathbf{x})) = \begin{cases} p_r^t(I_r(\mathbf{x})|\mathcal{F}) p_g^t(I_g(\mathbf{x})|\mathcal{F}) p_b^t(I_b(\mathbf{x})|\mathcal{F}) & \text{when} \quad \mathbf{x} \in \Omega^t \\ p_r^t(I_r(\mathbf{x})|\mathcal{B}) p_g^t(I_g(\mathbf{x})|\mathcal{B}) p_b^t(I_b(\mathbf{x})|\mathcal{B}) & \text{otherwise} \end{cases} \tag{4.11}$$

where I_r, I_g, I_b are intensity values in red, green, blue channels.

After document capture in the current frame, the optimal state vector $\hat{\omega}^t$ can be estimated and color distributions of document and background regions are updated for the next frame. The update criterion is a simple weighted sum as follows:

$$p_i^{t+1}(\cdot|l) = (1 - \alpha) p_i^t(\cdot|l) + \alpha h_i^t(\cdot|l) \tag{4.12}$$

where $h_i^t(\cdot|l)$ is a normalized intensity histogram of the pixel having a label $l \in \{\mathcal{F}, \mathcal{B}\}$ in $i \in \{r, g, b\}$ color channel.

Contrast term

Document boundaries usually lie on the high contrast regions and two kinds of contrast measures are proposed. First, given $\mathcal{L}^t$, the binary high contrast map C^t is constructed using line segments that the pixel close to the line segments is assigned to 1 and other pixels have 0 value. Then, the ratio of high contrast region on the

boundaries is evaluated. On the other hand, color contrast near each boundary is also calculated using histogram intersection. Finally, contrast energy is given by

$$E_C(\omega; \mathcal{L}^t) = \sum_{q_i^t \in Q^t} [-\frac{1}{|q_i^t|} \sum_{\mathbf{x} \in q_i^t} C^t(\mathbf{x}) + \gamma_1 H(q_i^t)] \tag{4.13}$$

$$H(q_i^t) = \sum_k \min(H_{1,q_i^t}(k), H_{2,q_i^t}(k)) \tag{4.14}$$

where H_{1,q_i^t} and H_{2,q_i^t} are normalized color histogram of both side around q_i^t and $|q_i^t|$ means the length of q_i^t.

Temporal term

For temporal coherency, two terms are designed that the variations of four corners and the ratio of length change should be small. In order to prevent the solution from converging to the wrong state, the bounded distance is adopted in the corner variation energy. The temporal term is defined as

$$\begin{aligned} E_T(\omega^t; \hat{\omega}^{t-1}) = & \sum_{q_i^t \in Q^t, q_i^{t-1} \in \hat{Q}^{t-1}} \left| \frac{|q_i^t|}{|q_i^{t-1}|} - 1 \right| + \\ & \gamma_2 \sum_{\mathbf{p}_i^t \in \omega^t, \mathbf{p}_i^{t-1} \in \hat{\omega}^{t-1}} \min(\max(\|\mathbf{p}_i^t - \mathbf{p}_i^{t-1}\| - B_0, B_1), B_2) \end{aligned} \tag{4.15}$$

where B_0, B_1, B_2 are constants.

4.4.4 Refinement

Since the best proposal $\tilde{\omega}^t$ depends on the line segments that are extracted on the highest contrast channel of the filtered image J^t, refinement is necessary for better representation. The edge features in the original image I^t are extracted using the Canny edge detector [43] and used for better localization. The boundary is refined so that the refinement boundaries pass through edge points as many as possible and refined state vector is represented as $\hat{\omega}^t$.

4.4.5 Experimental results

The proposed document capture algorithm is evaluated on a dataset from *ICDAR'2015 Competitions* [44] which consists of six different document types, and five document images are chosen per class. The 30 documents are printed on normal A4 paper, but they have different layout schemes that some have only textual contents and others include high graphical materials. Each document is recorded to small video clip in five different background scenarios using Full HD 1920×1080 resolution so that total test videos are 150 clips comprising around $25,000$ frames. Fig. 4.10 and 4.11 show the representative examples of six document types and five background scenarios, respectively. The document is easy to distinguish from the background in *background1* and *background2*, *background4* have light background that is similar to document color, *background3* has low lighting condition, and *background5* is very challenging scenario because of strong partial occlusions.

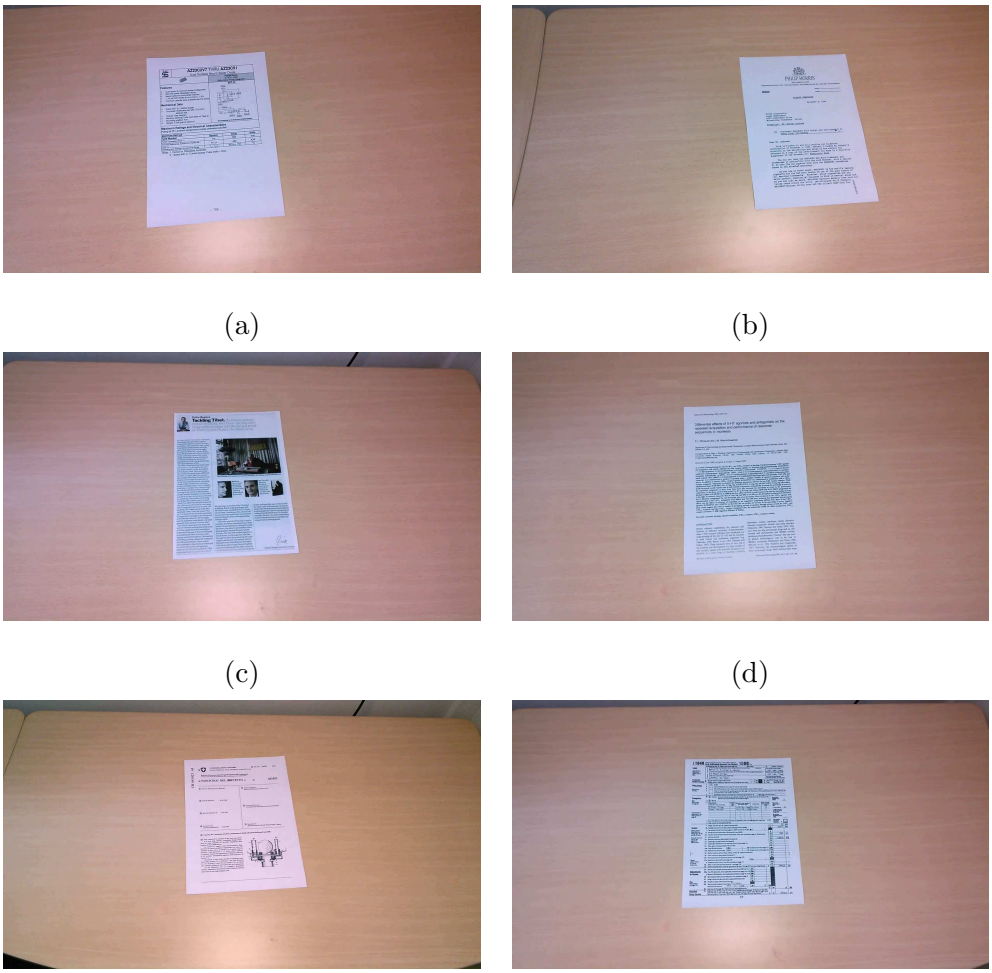
Implementation

In all experiments, distributions $p_i^t(\cdot|l)$ and $h_i^t(\cdot|l)$ are histograms with 32 bins, and $H_{1,q_i^t}(\cdot)$ and $H_{2,q_i^t}(\cdot)$ have $8 \times 8 \times 8$ bins. Other parameters are set empirically: $\lambda_D = 1$, $\lambda_C = 4$, $\lambda_T = 40$, $\gamma_1 = 2.5$, $\gamma_2 = 0.0025$, $B_0 = 10$, $B_1 = 0$, $B_2 = 25$, and $\alpha = 0.01$.

At the first frame, the simple indicator function is defined to assign the rough pixel label:

$$\text{ind}(\mathbf{x}) = \begin{cases} \mathcal{F} & \text{if} \quad \|HE(J(\mathbf{x}))\| > TH \\ \mathcal{B} & \text{otherwise} \end{cases} \tag{4.16}$$

where $HE(\cdot)$ means the histogram equalization operator and TH is set to 150. Then, the distributions $p_i^1(\cdot|l)$ are initialized with the rough pixel labels.

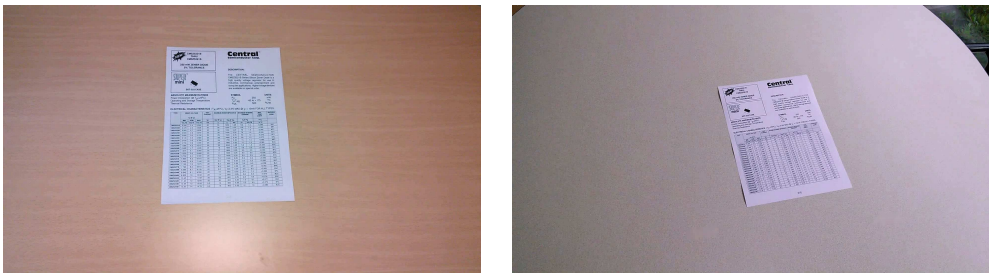


(a) (b)

(c) (d)

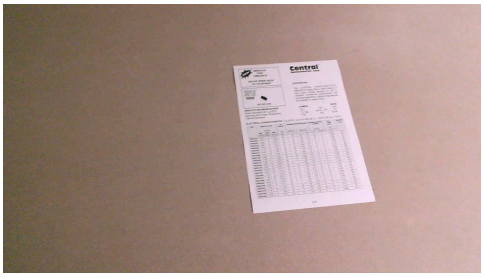
(e) (f)

Figure 4.10: Six document types. (a) datasheet, (b) letter, (c) magazine, (d) paper, (e) patent, (f) tax.

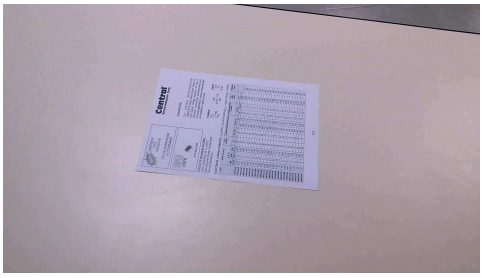


(a)

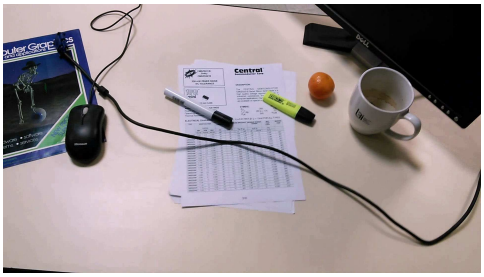
(b)



(c)



(d)



(e)

Figure 4.11: Five background scenarios. (a) background1, (b) background2, (c) background3, (d) background4, (e) background5.

Effect of filtering

In the situation of normal A4 papers, the preprocessing filtering (morphological close operation) is very useful to extract only line segments which lie on the document boundary. The texts, graphics and lines, which are regarded as the noise in the algorithm, are removed by the filtering while the document boundaries have still high contrast. Fig. 4.12 shows the examples of the filtering results.

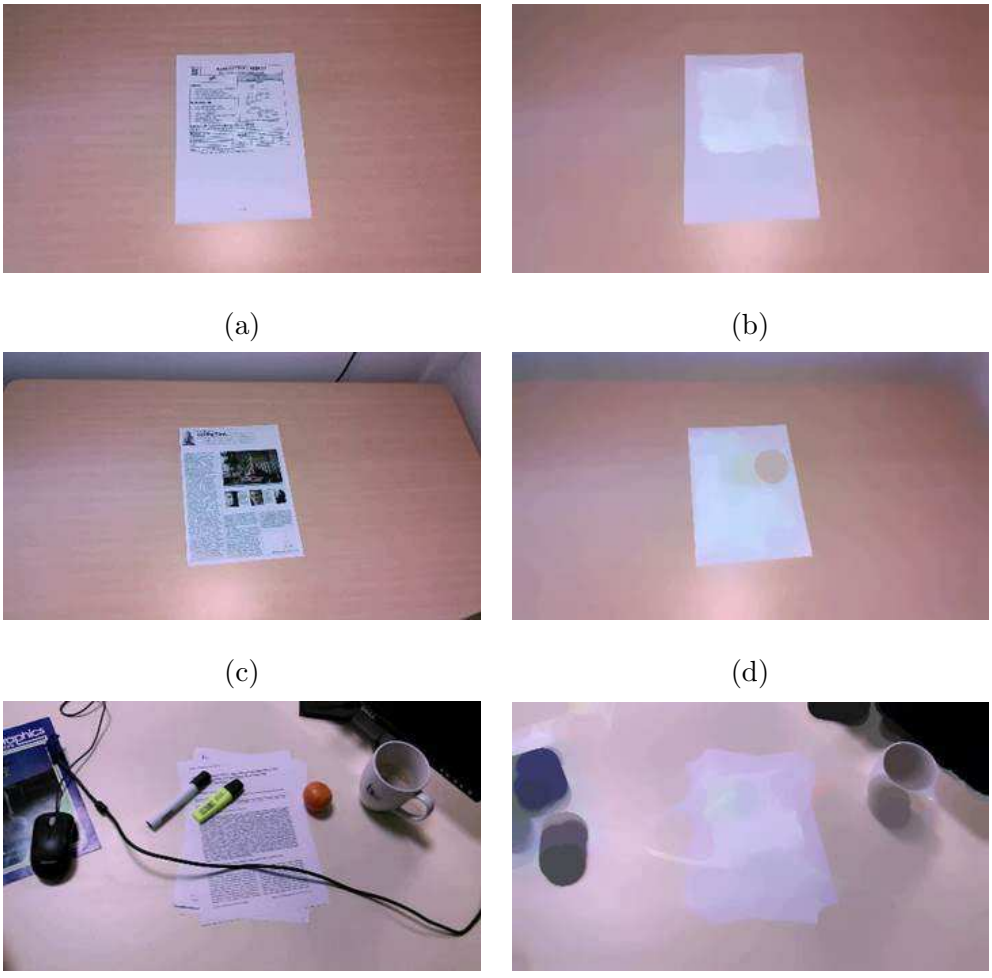
Also, the selected highest contrast color channel is superior to the normal gray image to extract line segments so that the document proposals are established robustly. The comparison of extracted line segments between the highest contrast channel and the gray image is illustrated in Fig. 4.13.

Comparison to other methods

For the evaluation of the proposed method, Jaccard index (JI) of document areas between estimated region and ground truth is computed :

$$\mathrm{JI}(\omega^t, \omega_g^t) = \frac{\bar{\Omega}^t \cap \bar{G}^t}{\bar{\Omega}^t \cup \bar{G}^t} \tag{4.17}$$

where ω_g^t represents the state of ground truth quadrilateral and its region is the G^t, $\bar{\Omega}^t$ and $\bar{G}^t$ are corrected rectangles of Ω^t and G^t respectively considering the document size and its coordinate using the perspective transform. When Ω^t corresponds to G^t exactly, JI score yields 1 and worst case gives 0 score. Ground truths are also given from *ICDAR'2015 Competitions* [44] and comparison results are summarized in Table 4.3, 4.4 and 4.5 where "ISPL-CVML" means the proposed method. As shown Table 4.3, the proposed method is ranked second in terms of mean score and confidence interval, however, average JI score of the proposed method is very similar to the best method (LRDE) whose algorithm is not published yet.

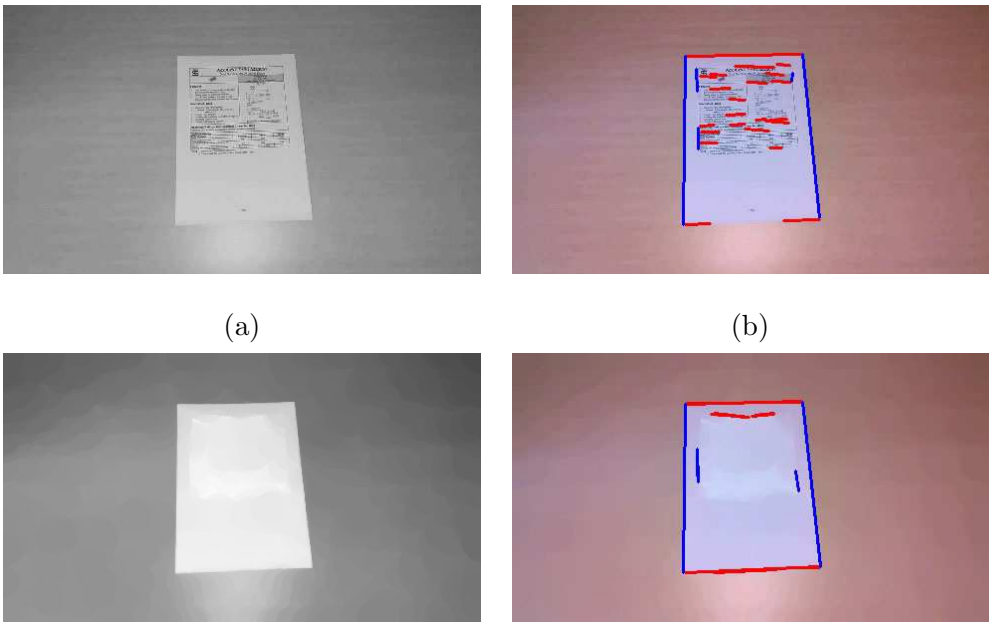


(a) (b)

(c) (d)

(e) (f)

Figure 4.12: The examples of the pre-filtering are showed.



(a) (b)

(c) (d)

Figure 4.13: The comparison of extracting line segments between the highest contrast channel and the gray image. Red line segments means horizontal lines and blue line segments stand for vertical ones. (a) gray image, (b) line segments from (a), (c) highest contrast channel image, (d) line segments from (c)

Table 4.3: The average JI score for overall test dataset.

Ranking	Method	Jaccard Index	Confidence Interval
1	LRDE	0.9716	[0.9710, 0.9721]
2	ISPL-CVML	0.9658	[0.9649, 0.9667]
3	SmartEngines	0.9548	[0.9533, 0.9562]
4	NetEase	0.8820	[0.8790, 0.8850]
5	A2iA run 2	0.8090	[0.8049, 0.8132]
6	A2iA run 1	0.7788	[0.7745, 0.7831]
7	RPPDI-UPE	0.7408	[0.7359, 0.7456]
8	SEECS-NUST	0.7393	[0.7353, 0.7432]

On the other hand, the average JI scores are measured and analyzed per document types and different background scenarios in Table 4.4 and Table 4.5. First, the performance against document types demonstrates that the average score is generally low when the document has complicated constitution with many lines and graphical materials, however, the deviation between the worst and the best is small in the proposed method. Second, the document capture system is more affected by the the light background than low light condition and severe occlusion is very challenging in finding the document boundaries. As shown Table 4.4, the proposed method captures the document boundary in different background scenarios, in addition, the average JI score yields higher score than other method in the situation of partial severe occlusion.

Table 4.4: The average JI score per background. The best results are in boldface.

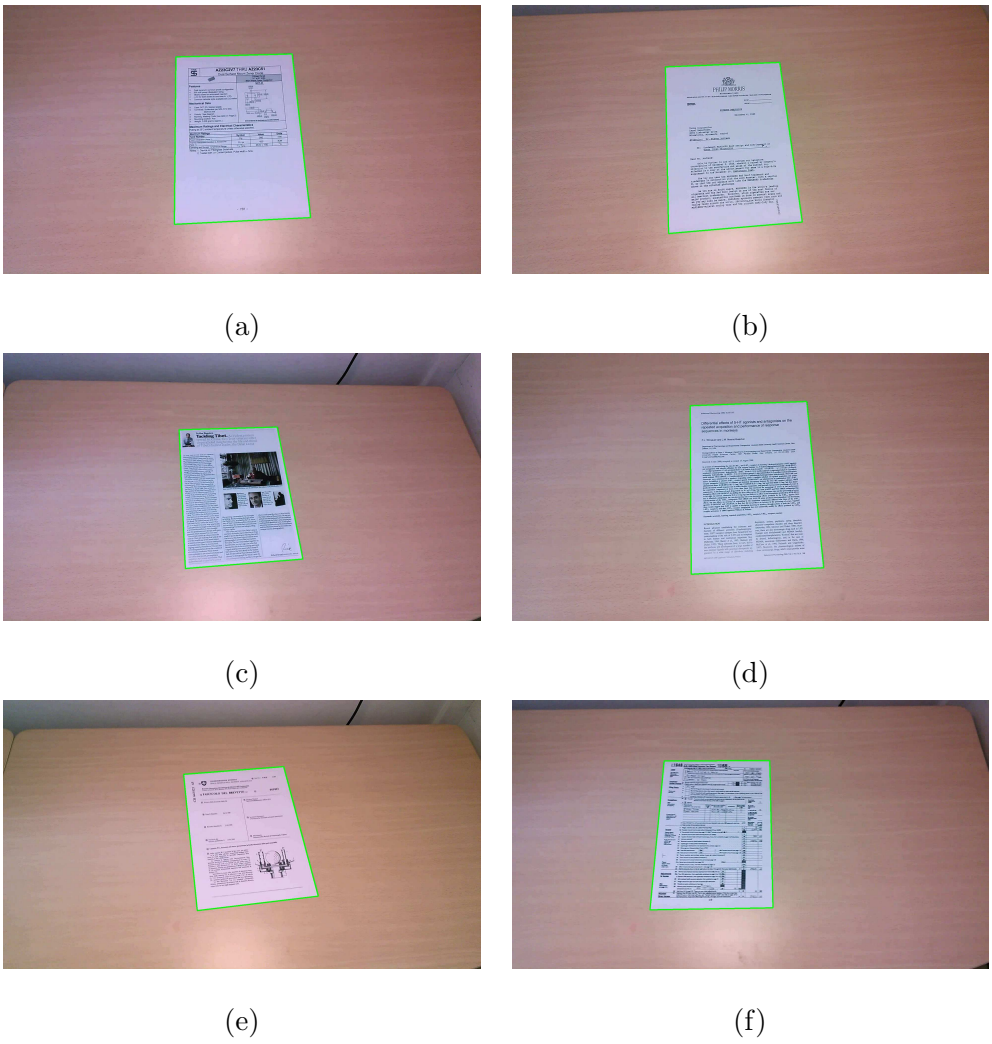
Method	Background				
	1	2	3	4	5
A2iA-1	0.9724	0.8006	0.9117	0.6352	0.1890
A2iA-2	0.9597	0.8063	0.9118	0.8264	0.1892
ISPL-CVML	0.9870	0.9652	0.9846	0.9766	0.8555
LRDE	0.9869	0.9775	0.9889	**0.9837**	**0.8613**
NetEase	0.9624	0.9552	0.9621	0.9511	0.2218
SEECS-NUST	0.8875	0.8264	0.7832	0.7811	0.0113
RPPDI-UPE	0.8274	0.9104	0.9697	0.3649	0.2163
SmartEngines	**0.9885**	**0.9833**	**0.9897**	0.9785	0.6884
All	0.9465	0.9031	0.9377	0.8122	0.4041

Table 4.5: The average JI score per document types. The best results are in boldface.

Method	Document types					
	datasheet	letter	magazine	paper	patent	tax
A2iA-1	0.8245	0.8005	0.7026	0.8555	0.7774	0.7073
A2iA-2	0.8538	0.8250	0.7577	0.8821	0.8060	0.7240
ISPL-CVML	**0.9761**	0.9691	0.9558	**0.9719**	0.9586	0.9626
LRDE	0.9758	**0.9718**	**0.9707**	0.9715	**0.9698**	**0.9696**
NetEase	0.8950	0.8666	0.8958	0.8798	0.8723	0.8817
SEECS-NUST	0.7745	0.8035	0.7292	0.7186	0.7470	0.6552
RPPDI-UPE	0.6606	0.7126	0.8232	0.7547	0.7191	0.7803
SmartEngines	0.9671	0.9498	0.9438	0.9596	0.9562	0.9517
All	0.8659	0.8624	0.8474	0.8742	0.8508	0.8290

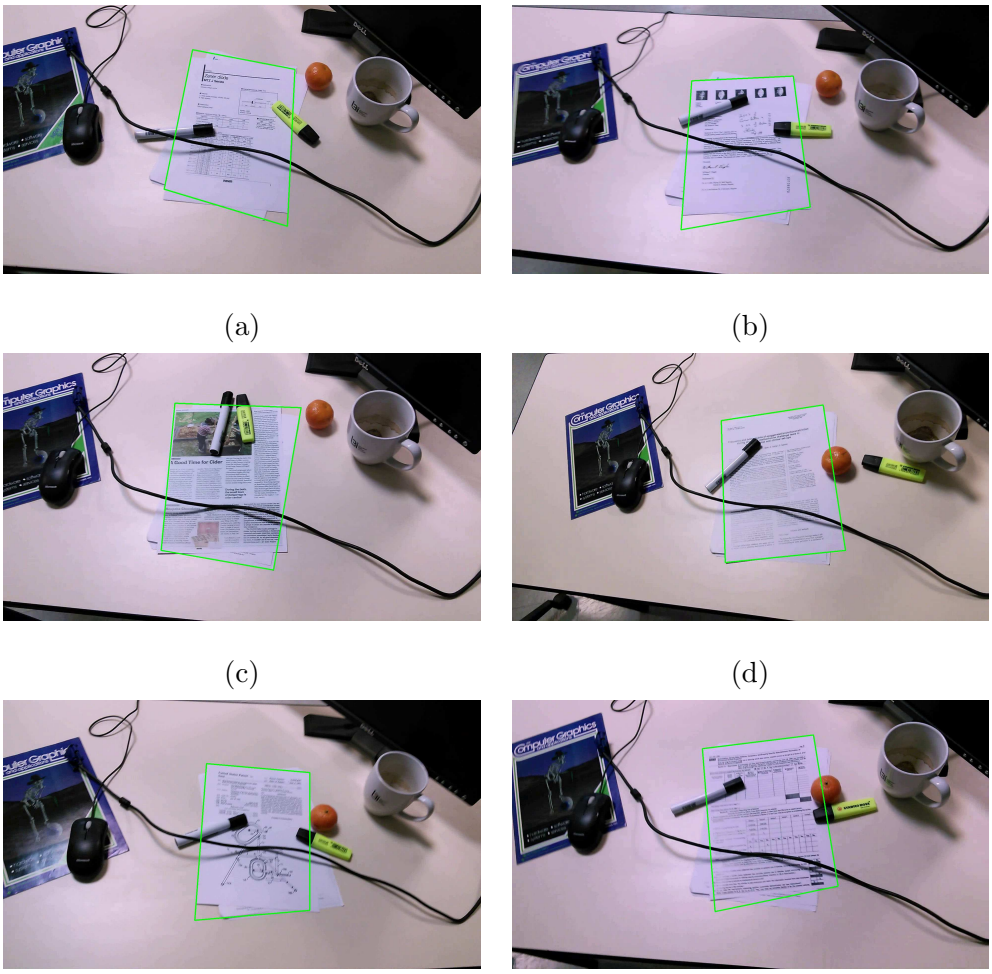
Qualitative evaluation and limitations

The results of document capture are illustrated in Fig. 4.14 and 4.15 for some test video sequences which have six document classes and five background scenarios. The proposed capture system depends on the line segment detector [42] which extracts line segments detecting high contrast regions so that it is difficult to find line segments when there has no contrast around document boundaries. Also, if severe occlusions exist as in *background5*, the proposed method may detect false document boundary as shown Fig. 4.15, however, this algorithm is robust system to capture normal documents as shown Table 4.3 and Fig. 4.14 $\sim$ 4.15.



(a) (b) (c) (d) (e) (f)

Figure 4.14: The document capture results of the proposed method in the *background1* are showed green quadrilateral. (a) datasheet, (b) letter, (c) magazine, (d) paper, (e) patent, and (f) tax.



(a) (b) (c) (d) (e) (f)

Figure 4.15: The document capture results of the proposed method in the *background5* are showed green quadrilateral. (a) datasheet, (b) letter, (c) magazine, (d) paper, (e) patent, and (f) tax.

Chapter 5

Multi-contours tracking for objects that belong to the same category

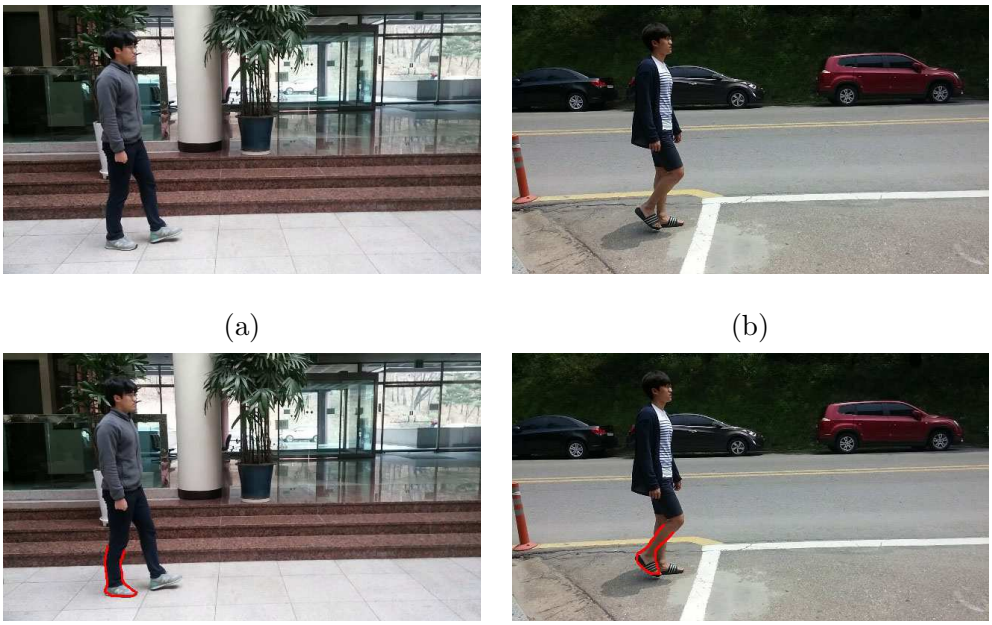
5.1 Proposed multi-contours tracking

If target objects belong to the same category, then it is hard to track each object correctly since all objects have similar shapes, colors and appearances. A tracked target may be hijacked to another target due to similar characteristics between them. For example, human legs are in the same class and resemble each other, thus they are chosen as targets to be tracked in this chapter. The proposed method adopts the same framework with the open contour tracking in Chapter 3 to solve the tracking problem of multi-contours that have similar appearances each other. Multi-contours are some boundary parts of objects and modeled together with one state vector and they are tracked with one tracker based on the recursive Bayesian

estimation [15] in (1.1). It considers the interactions between contours so that we can avoid the contour overlapping problem which means that some contours coincide at the same location where the probability of being target is very high. However, there exists some problems that the state space is large and complex (the dimension of state vector is high) and a great number of particles are necessary. Therefore, it is difficult and time consuming to find the optimized state vector because it should cover large state space. The proposed method provides an efficient manner to search the optimum state with numerable particles.

5.1.1 State space model

At time t, the state vector for one contour consists of rigid transformation matrix $\mathbf{X}^t$ and non-rigid deformation coefficients $\mathbf{q}^t$: $\boldsymbol{\omega}_i^t = \{\mathbf{X}_i^t, \mathbf{q}_i^t\}$ and the state vector for multi-contours is simply constructed by concatenating $\boldsymbol{\omega}_i^t$: $\boldsymbol{\omega}^t = \{\boldsymbol{\omega}_1^t, \boldsymbol{\omega}_2^t\}$. Therefore, interactions between contours can be taken into account using the state vector and the tracker is able to yield more accurate results, however, the state space is larger and more complex than the case of single contour. Control points $\mathbf{D}_i^t$ of each contour are calculated with rigid and non-rigid motions $\boldsymbol{\omega}_i^t$, the mean shape $\bar{\mathbf{D}}$ and non-rigid motion basis $\mathbf{W}$ as mentioned in (3.5) where $\bar{\mathbf{D}}$ and $\mathbf{W}$ are established through the training process (Algorithm 1) before the tracking. Since all objects belong to the same category, $\bar{\mathbf{D}}$ and $\mathbf{W}$ can be employed to all contours simultaneously. Finally, each contour $\mathbf{r}_i^t(\cdot)$ is realized based on the active contour model (B-spline model) in (3.2). We utilize aperiodic cubic B-splines because contours are open. The procedure of obtaining ground truth contour, estimating the state vector and drawing the contour $\mathbf{r}_i^t(\cdot)$ with its control points $\mathbf{D}^t$ is developed in Fig. 5.1.



(a) (b)

(c) (d)

(e) (f)

Figure 5.1: First row: input images. Second row: ground truth contours that are drawn manually. Third row: estimated contours $\mathbf{r}(u)$ (red curve) and their control points $\mathbf{D}$ (blue circles).

5.1.2 Dynamics and measurement

According to the recursive Bayesian estimation (1.1), the state vector $\boldsymbol{\omega}^t$ is moved depending on the dynamic model $p(\boldsymbol{\omega}^t|\boldsymbol{\omega}^{t-1})$ where each contour proceeds respectively. Furthermore, we should consider the interactions between contours so that geometric relations make sense. Therefore, we add the interaction term to independent dynamic terms:

$$p(\boldsymbol{\omega}^t|\boldsymbol{\omega}^{t-1}) = p(\boldsymbol{\omega}_1^t|\boldsymbol{\omega}_1^{t-1})p(\boldsymbol{\omega}_2^t|\boldsymbol{\omega}_2^{t-1})p(\boldsymbol{\omega}_1^t, \boldsymbol{\omega}_2^t). \tag{5.1}$$

The interaction term is designed to apply three ideas that contours prefer to be apart each other, geometric relations of them have to be similar and their motions comply with the law of inertia. First, we give the penalty when contours are overlapped (high overlap score). The overlap score is evaluated using Jaccard index between object regions $J_{obj}(\boldsymbol{\omega}_i^t)$ which are made with the rule that the area of "object" label in the label map $J(\boldsymbol{\omega}_i^t)$ is regarded as the object region. The probability of avoiding overlap is defined as follow equation:

$$p_{overlap}(\boldsymbol{\omega}_1^t, \boldsymbol{\omega}_2^t) \propto \exp(-\lambda_{ov} \times \mathrm{JI}(J_{obj}(\boldsymbol{\omega}_1^t), J_{obj}(\boldsymbol{\omega}_2^t)) \tag{5.2}$$

where $\mathrm{JI}(\cdot,\cdot)$ means the JI score as mentioned in (3.47) and λ_{ov} is a constant. Second, the sizes of contours should be similar and the distance between them should be smaller than a certain threshold that is proportional to their sizes. The probability of geometric relation is designed as

$$p_{geo}(\boldsymbol{\omega}_1^t, \boldsymbol{\omega}_2^t) \propto \quad \exp\left(-\lambda_{g1}\frac{2|s_1^t - s_2^t|}{s_1^t + s_2^t} - \lambda_{g2} \times h\left(\|\frac{1}{n}\mathbf{D}_1^t\mathbf{1}_{n\times 1} - \frac{1}{n}\mathbf{D}_2^t\mathbf{1}_{n\times 1}\|\right)\right) \tag{5.3}$$

$$h(x) = \begin{cases} 0 & \text{when} \quad 0 \le x < T_1 \\ \frac{1}{T_2 - T_1}(x - T_1) & \text{when} \quad T_1 \le x < T_2 \\ 1 & \text{when} \quad x \ge T_2 \end{cases} \tag{5.4}$$

where s_i^t presents the size of i-th contour as denoted in (3.5) and (3.6), $\frac{1}{n}\mathbf{D}_i^t\mathbf{1}_{n\times 1}$ stands for the center of control points $\mathbf{D}_i^t$, T_1 and T_2 depend on the scale s^t, and λ_{g1} and λ_{g2} are constants. Finally, in order to apply the law of inertia, we estimate the center location $\mathbf{c}_i^t$ of i-th contour and examine distances from $\mathbf{c}_i^t$ to centers of all contours. The distance from $\mathbf{c}_i^t$ to its corresponding center $\frac{1}{n}\mathbf{D}_i^t\mathbf{1}_{n\times 1}$ should be the smallest one and we implement it in a distinctive manner. The i-th state vector is determined to the one that releases the smallest distance from $\mathbf{c}_i^t$:

$$\boldsymbol{\omega}_i^t = \arg\min_{\boldsymbol{\omega}_j^t} \|\mathbf{c}_i^t - o(\boldsymbol{\omega}_j^t)\| \tag{5.5}$$

where $o(\boldsymbol{\omega}_j^t)$ means the center of control points determined by $\boldsymbol{\omega}_j^t$ and $\mathbf{c}_i^t$ is approximated by the adding velocity vector $\mathbf{v}_i^t$ which is updated every frame with auto regression model:

$$o(\boldsymbol{\omega}_j^t) = \frac{1}{n}\left(s^t\mathbf{R}^t(\bar{\mathbf{D}} + \text{reshape}(\mathbf{W}\mathbf{q}^t)) + \mathbf{p}^t\mathbf{1}_{n\times 1}^\top\right)\mathbf{1}_{n\times 1} \tag{5.6}$$

$$\mathbf{c}_i^t = \frac{1}{n}\mathbf{D}_i^{t-1}\mathbf{1}_{n\times 1} + \mathbf{v}_i^t \tag{5.7}$$

$$\mathbf{v}_i^{t+1} = (1-\alpha_v)\mathbf{v}_i^t + \alpha_v\left(\frac{1}{n}\mathbf{D}_i^t\mathbf{1}_{n\times 1} - \frac{1}{n}\mathbf{D}_i^{t-1}\mathbf{1}_{n\times 1}\right) \tag{5.8}$$

where α_v is a constant. In summary, the interaction term is defined as

$$p(\boldsymbol{\omega}_1^t, \boldsymbol{\omega}_2^t) = p_{overlap}(\boldsymbol{\omega}_1^t, \boldsymbol{\omega}_2^t) \times p_{geo}(\boldsymbol{\omega}_1^t, \boldsymbol{\omega}_2^t), \tag{5.9}$$

then $\boldsymbol{\omega}_i^t$ is optimized depending on (5.5). Actually, the interaction term $p(\boldsymbol{\omega}_1^t, \boldsymbol{\omega}_2^t)$ is calculated at the time of evaluating the measurement in our implementation. Other terms of (5.1) are same to (3.7) as mentioned in Section 3.1.3. Also, measurement terms are equivalent to the definitions in Section 3.3.

5.1.3 Particle sampling

Since multi-contours are modeled by a single state vector simultaneously, the state vector space is complex and large, and many particles are required to reflect diverse combinations of multi-contours. This may not be feasible work owing to time and memory problems. Thus, we provide an efficient way to draw competitive particle samples with the fixed number of particles. We divide the process into two steps that each contour is evolved independently and particles are sorted as an ascending order of energy that is derived from the measurement in the first step, then we select combinations that have high probabilities including the interaction term in the other step. Several particles are chosen at random regardless of their probabilities in order to prevent falling into the wrong local minimum and increase robustness. To be precise, particles of each contour are moved based on the dynamic model:

$$\boldsymbol{\omega}_i^{t,(j)} \sim p(\boldsymbol{\omega}_i^t | \boldsymbol{\omega}_i^{t-1,(j)}) \tag{5.10}$$

where $\boldsymbol{\omega}_i^{t,(j)}$ presents j-th particle of i-th contour, and their energies are calculated by measurement terms E_C, E_K and E_A:

$$E(\boldsymbol{\omega}_i^{t,(j)}) = \lambda_C E_C(\boldsymbol{\omega}_i^{t,(j)}) + \lambda_K E_K(\boldsymbol{\omega}_i^{t,(j)}) + \lambda_A E_A(\boldsymbol{\omega}_i^{t,(j)}). \tag{5.11}$$

Then, N_ψ particles that have N_ψ lowest energies are collected to a set Ψ_i^t:

$$\Psi_i^t = \{\boldsymbol{\psi}_i^{t,(1)}, \boldsymbol{\psi}_i^{t,(2)}, \cdots, \boldsymbol{\psi}_i^{t,(N_\psi)}\} \tag{5.12}$$

where $\boldsymbol{\psi}_i^{t,(j)}$ is the state vector of i-th contour and Ψ_i^t is called a superior particle set of i-th contour at time t. Next, j-th element $\boldsymbol{\psi}_1^{t,(j)}$ is matched to $\rho(j)$-th element $\boldsymbol{\psi}_2^{t,(\rho(j))}$ that these combination yields the smallest energy where $\rho(j)$ can be

estimated as following equation:

$$\rho(j) = \arg\min_{\rho(j)} E(\boldsymbol{\psi}_1^{t,(j)}) + E(\boldsymbol{\psi}_2^{t,(\rho(j))}) - \log\left(p(\boldsymbol{\psi}_1^{t,(j)}, \boldsymbol{\psi}_2^{t,(\rho(j))})\right). \tag{5.13}$$

Estimated N_ψ combinations are set to N_ψ particles of the state vector $\boldsymbol{\omega}^t$ and the rest particles are determined by joining two state vectors which are chosen randomly from each state vector set. These particles assist the tracker to avoid converging on the wrong local minimum and increase the performance. Finally, drawn particles are optimized using (5.5).

5.2 Experimental results

For the training process, 131 leg shapes are produced manually and then the mean shape $\bar{\mathbf{D}}$ and the non-rigid basis $\mathbf{W}$ are guessed simultaneously. We can verify the effect of non-rigid basis in Fig. 5.2 where we provide shape examples constructed by (3.45).

The number of control points n and the number of non-rigid motion coefficients k are selected by measuring the Chamfer distance similar to Section 3.4.1 and 4.3. $n = 13$ and $k = 2$ are selected where the Chamfer distances are less than 1 pixel for the first time as shown Fig. 5.3. Weight parameters of interaction term are set to $\lambda_{ov} = 20$, $\lambda_{g1} = 15$, $\lambda_{g2} = 50$, the range parameters in (5.4) are calculated as $T_1 = 1.2s^t, T_2 = 2.2s^t$ and the ratio of updating velocity vector is determined as $\alpha_v = 0.7$. We adopt the adaptive scheme for measurement terms as mentioned in Section 3.3.5 and other parameters are also same to the experiment setting in Section 3.4.1. We decide the number of elements in Ψ_i^t as a tenth of the total particle number that $N_\psi = 60$ and 600 particles are used for multi-contours tracking.

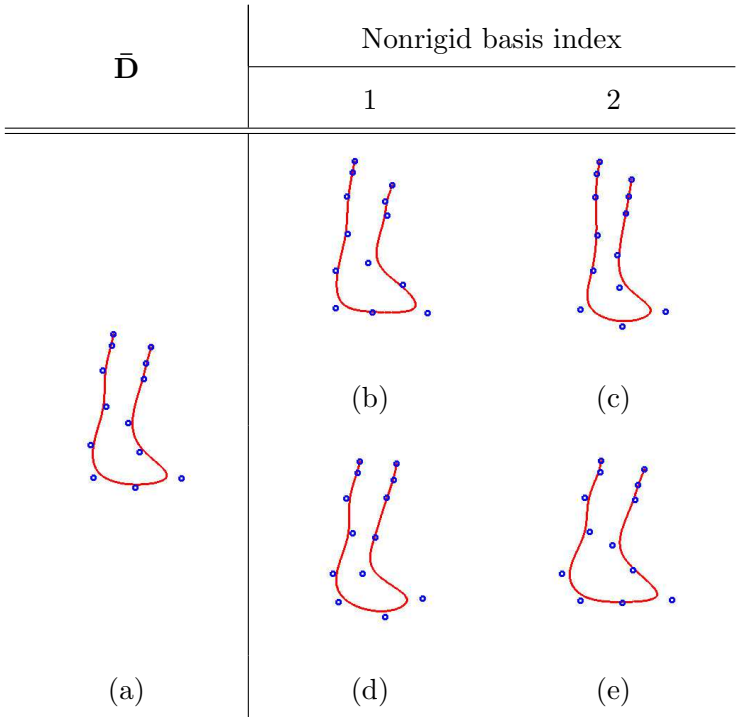
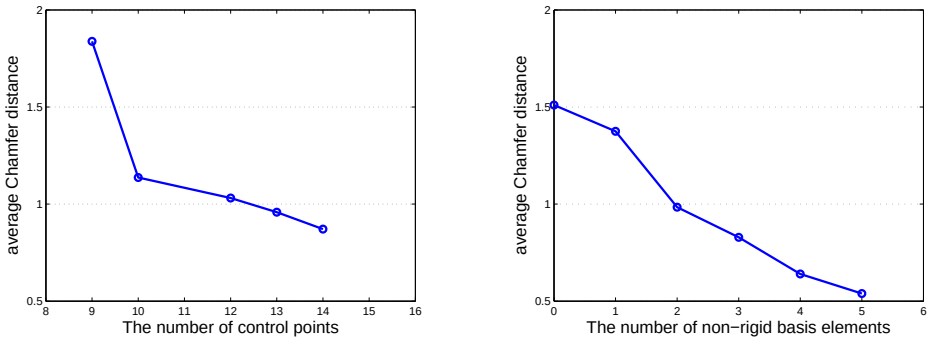


Figure 5.2: Illustration of nonrigid basis vectors for a leg shape: (a) mean shape $\bar{\mathbf{D}}$, (b)-(d) contours corresponding to $\bar{\mathbf{D}} + \text{reshape}(\mathbf{W}(0, \cdots, 0, 2\sigma_i, 0, \cdots, 0)^\top)$, (e)-(g) contours corresponding to $\bar{\mathbf{D}} - \text{reshape}(\mathbf{W}(0, \cdots, 0, 2\sigma_i, 0, \cdots, 0)^\top)$.



(a) (b)

Figure 5.3: Chamfer distances for varying parameters: (a) Chamfer distances from ground truth to reconstructed contours for several n (the number of control points), (b) Chamfer distances from original B-spline contours to the contours reconstructed with k basis vectors.

(a) (b)

Figure 5.4: The label map $J(\boldsymbol{\omega}_i^t)$ examples are demonstrated. The intensity in $J(\boldsymbol{\omega}_i^t)$ are the label and black pixels present "don't care" region.

Also, we adopt three labels for a leg shape: $\mathcal{L} = \{$"object", "background", "don't care"$\}$ and the label map $J(\boldsymbol{\omega}_i^t)$ is constructed by connecting two end points of the open contour, that is the same way of cheetah profile case in Section 3.4.2. Label map examples are demonstrated in Fig. 5.4.

5.2.1 Comparison with other multi-objects tracking methods

Since human legs are the targets in this study, the proposed method is tested on three sequences where a person walks in various backgrounds. It may be challenging to track legs accurately because human legs cross every steps and one leg cannot help being occluded by the other during walking. Moreover, two legs have the same shapes, sizes, colors and appearances, and background color is also very similar to legs' color in "walk2" sequence. These factors make it difficult to track multi-objects simultaneously. Some example frames of test sequences are provided in Fig. 5.5.

Most conventional multi-objects tracking methods conduct the object detector and they focus on the data association problem of detected objects between frames. Humans are chosen as their targets and results of human detector are essential to work algorithms. In order to compare the proposed method with this approach, [35] is selected because all parts of human including legs are detected using [35]. In addition, the proposed method is compared with [36] where multi-objects share the same model and tracked by MCMC-based method. Since conventional methods represent the target as closed contours (rectangles), the JI score in (3.47) is measured and A_{est} (object region) of the proposed method can be constructed easily with the label map $J(\boldsymbol{\omega}_i^t)$. The ratio of successfully tracked frames is also estimated according to the JI score. When the JI score is greater than a certain threshold (we use 0.5), the leg is regarded to tracked correctly. If all legs are estimated well, then the


(a) walk1


(b) walk2


(c) walk3

Figure 5.5: Representative frames of test sequences are presented.

Table 5.1: Comparison with conventional multi-objects tracking methods for the ratio of successfully tracked frames. The best results are in boldface.

Sequences	MCMC [36]	Articulated pose [35]	Proposed
walk1	0.043	0.087	**0.957**
walk2	0.071	0.071	**0.929**
walk3	0.048	0.048	**0.714**
average	0.054	0.069	**0.867**

Table 5.2: Comparison with conventional multi-objects tracking methods for the average JI score. The best results are in boldface.

Sequences	MCMC [36]	Articulated pose [35]	Proposed
walk1	0.305	0.299	**0.730**
walk2	0.147	0.194	**0.655**
walk3	0.209	0.210	**0.611**
average	0.220	0.234	**0.665**

frame is considered to the success. Table 5.1 and 5.2 show the results of multi-legs tracking. The proposed method achieves better performances and conventional methods have trouble to track legs and distinguish them exactly. Also, the average Chamfer distances in (3.46) are measured for results of the proposed method in order to represent accuracies of multi-contours. As shown in Table 5.3, the proposed method estimates multi-contours with high precisions. Fig. 5.6, 5.7 and 5.8 present illustration results for the proposed method and [35, 36].

Table 5.3: The average Chamfer distance of the proposed method.

Sequences	first leg	second leg	average
walk1	1.305	1.482	1.544
walk2	1.590	1.734	1.662
walk3	3.120	2.617	2.869
average			2.025

(a) MCMC [36]

(b) Articulated pose [35]

(c) Proposed

Figure 5.6: Tracking results of (a) MCMC [36], (b) Articulated pose [35] and (c) the proposed method are represented in "walk1" sequence. Red and yellow colors are chosen to distinguish legs.

(a) MCMC [36]

(b) Articulated pose [35]

(c) Proposed

Figure 5.7: Tracking results of (a) MCMC [36], (b) Articulated pose [35] and (c) the proposed method are represented in "walk2" sequence. Red and yellow colors are chosen to distinguish legs.

(a) MCMC [36]

(b) Articulated pose [35]

(c) Proposed

Figure 5.8: Tracking results of (a) MCMC [36], (b) Articulated pose [35] and (c) the proposed method are represented in "walk3" sequence. Red and yellow colors are chosen to distinguish legs.

Table 5.4: Comparison with conventional single object tracking methods for the average JI score. The best results are in boldface.

Sequences	IVT [2]	CSK [4]	Proposed
walk1	0.232	0.212	**0.730**
walk2	0.003	0.204	**0.655**
walk3	0	0.173	**0.611**
average	0.078	0.196	**0.665**

5.2.2 Comparison with tracking methods for a single object

The proposed method is compared to conventional tracking methods [2, 4] which are introduced in previous chapters (Chapter 3 and 4) and developed for a single object. As mentioned in Section 5.2.1, human legs have the same appearances and tracking may fail when they cross each other. The JI score in (3.47) is adopted to measure the performance. The results are represented in Table 5.4 and illustration examples are demonstrated in Fig. 5.9. Conventional methods for a single object [2, 4] do not recognize the proper target (left leg in Fig. 5.9) because very similar object (right leg) exists close to the target. On the other hand, the proposed method chases all legs correctly and yields high JI scores.

(a) “walk1”

(a) “walk2”

(a) “walk3”

Figure 5.9: Tracking results of conventional single object tracking methods and the proposed method are represented in the sequence of (a) “walk1”, (b) “walk2” and (c) “walk3”. First column means the first frame of each sequence. Single object tracking methods fail to track the target (the left leg). Green, blue and red curves correspond to IVT [2], CSK [4] and the proposed method.

Chapter 6

Conclusions

In this dissertation, a new open and closed contours tracking method based on shape priors and their training is proposed. A part or the whole of the object's boundary is represented by a contour (open or closed) and it is tracked by the recursive Bayesian estimation framework. The proposed state space model can describe any contours regardless of the shape and handle rigid and non-rigid motions independently. According to the model, the proposed method focuses on the non-rigid motions which are trained with the proposed learning criterion in advance. Though the non-rigid movements are trained only, the proposed method adopts well known rigid motion models and this enables to work with challenging rigid motions. Also, the measurement terms are designed in consideration of the contrast on object's boundary, target appearance and temporal coherency. The proposed method is applied to diverse targets such as human omega shape, a cheetah profile, car shapes shown by the black box, a document shape and human legs. Numerous experiments are carried out on several sequences for every targets and the proposed method achieves better performances compared to conventional tracking methods. The results of multi-

contours tracking also shows that the proposed method can distinguish targets that are similar to each other. Besides, the proposed method can estimate boundaries of targets as well as their locations and it will be helpful to recognize the real object region and pose. Therefore, the proposed method can be used as a preprocessing step for other computer vision algorithms like elaborate segmentations or pose estimations. In summary, this dissertation proposes a new algorithm to estimate open and closed contours using shape prior and the training criterion for target shapes is also provided. The rigid and non-rigid motions are separated definitely and the proposed method achieves plausible tracking performances compared with conventional methods.

Bibliography

[1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Comput. Surv.*, vol. 38, no. 4, Dec. 2006.

[2] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 125–141, 2008.

[3] S. Oron, A. Bar-Hille, and S. Avidan, "Extended lucas-kanade tracking," in *European Conference on Computer Vision*. Springer, 2014, pp. 142–156.

[4] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," in *European Conference on Computer Vision*. Springer, 2012, pp. 702–715.

[5] K. Zhang, L. Zhang, and M.-H. Yang, "Fast compressive tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 10, pp. 2002–2015, 2014.

[6] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 983–990.

[7] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 7, pp. 1409–1422, 2012.

[8] J. Kwon, K. M. Lee, and F. C. Park, "Visual tracking via geometric particle filtering on the affine group with optimal importance functions," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 991–998.

[9] S. Wang, H. Lu, F. Yang, and M.-H. Yang, "Superpixel tracking," in *International Conference on Computer Vision*. IEEE, 2011, pp. 1323–1330.

[10] Z. Cai, L. Wen, Z. Lei, N. Vasconcelos, and S. Z. Li, "Robust deformable and occluded object tracking with dynamic graph," *IEEE Transactions on Image Processing*, vol. 23, no. 12, pp. 5497–5509, 2014.

[11] S. S. Nejhum, J. Ho, and M.-H. Yang, "Online visual tracking with histograms and articulating blocks," *Computer Vision and Image Understanding*, vol. 114, no. 8, pp. 901–914, 2010.

[12] J. Kwon and K. M. Lee, "Highly nonrigid object tracking via patch-based dynamic appearance modeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 10, pp. 2427–2441, 2013.

[13] Y. Rathi, N. Vaswani, A. Tannenbaum, and A. Yezzi, "Particle filtering for geometric active contours with application to tracking moving and deforming objects," in *Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2005, pp. 2–9.

[14] P. Chockalingam, N. Pradeep, and S. Birchfield, "Adaptive fragments-based tracking of non-rigid objects using level sets," in *International Conference on Computer Vision*. IEEE, 2009, pp. 1530–1537.

[15] M. Isard and A. Blake, "Condensation conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.

[16] X. Bai, J. Wang, D. Simons, and G. Sapiro, "Video snapcut: robust video object cutout using localized classifiers," *ACM Transactions on Graphics*, vol. 28, no. 3, p. 70, 2009.

[17] M. Godec, P. M. Roth, and H. Bischof, "Hough-based tracking of non-rigid objects," *Computer Vision and Image Understanding*, vol. 117, no. 10, pp. 1245–1256, 2013.

[18] M. Li, C. Kambhamettu, and M. Stone, "Automatic contour tracking in ultrasound images," *Clinical linguistics & phonetics*, vol. 19, no. 6-7, pp. 545–554, 2005.

[19] M. E. Sargin, A. Altinok, B. S. Manjunath, and K. Rose, "Variable length open contour tracking using a deformable trellis," *IEEE Transactions on Image Processing*, vol. 20, no. 4, pp. 1023–1035, 2011.

[20] T. Baltrušaitis, P. Robinson, and L.-P. Morency, "3d constrained local model for rigid and non-rigid facial tracking," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 2610–2617.

[21] S. Bouaziz, Y. Wang, and M. Pauly, "Online modeling for realtime facial animation," *ACM Transactions on Graphics*, vol. 32, no. 4, p. 40, 2013.

[22] V. Caselles, F. Catté, T. Coll, and F. Dibos, "A geometric model for active contours in image processing," *Numerische Mathematik*, vol. 66, no. 1, pp. 1–31, 1993.

[23] R. Malladi, J. A. Sethian, and B. C. Vemuri, "Shape modeling with front propagation: A level set approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 2, pp. 158–175, 1995.

[24] A.-R. Mansouri, "Region tracking via level set pdes without motion computation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 947–961, 2002.

[25] Y. Shi and W. C. Karl, "Real-time tracking using level sets," in *Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2005, pp. 34–41.

[26] B. Ma and Y. Wu, "Covariance matching for pde-based contour tracking," in *International Conference on Image and Graphics*. IEEE, 2011, pp. 720–725.

[27] X. Zhou, X. Li, T.-J. Chin, and D. Suter, "Superpixel-driven level set tracking," in *International Conference on Image Processing*. IEEE, 2012, pp. 409–412.

[28] C. Rother, V. Kolmogorov, and A. Blake, "Grabcut: Interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 309–314, 2004.

[29] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool, "Online multiperson tracking-by-detection from a single, uncalibrated camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1820–1833, 2011.

[30] J. Berclaz, F. Fleuret, E. Türetken, and P. Fua, "Multiple object tracking using k-shortest paths optimization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1806–1819, 2011.

[31] A. Andriyenko, K. Schindler, and S. Roth, "Discrete-continuous optimization for multi-target tracking," in *Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 1926–1933.

[32] H. Ben Shitrit, J. Berclaz, F. Fleuret, and P. Fua, "Multi-commodity network flow for tracking multiple people," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1614–1627, 2014.

[33] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Conference on Computer Vision and Pattern Recognition*, vol. 1. IEEE, 2005, pp. 886–893.

[34] B. Leibe, A. Leonardis, and B. Schiele, "Robust object detection with interleaved categorization and segmentation," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 259–289, 2008.

[35] Y. Yang and D. Ramanan, "Articulated human detection with flexible mixtures of parts," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2878–2890, 2013.

[36] Z. Khan, T. Balch, and F. Dellaert, "Mcmc-based particle filtering for tracking a variable number of interacting targets," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1805–1819, 2005.

[37] T. Zhao, R. Nevatia, and B. Wu, "Segmentation and tracking of multiple humans in crowded environments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 7, pp. 1198–1211, 2008.

[38] A. Blake and M. Isard, *Active Contours*. Springer, 1998, the Application of Techniques from Graphics, Vision, Control Theory and Statistics to Visual Tracking of Shapes in Motion.

[39] J. C. Gower, "Generalized procrustes analysis," *Psychometrika*, vol. 40, no. 1, pp. 33–51, 1975.

[40] E. S. Gastal and M. M. Oliveira, "Domain transform for edge-aware image and video processing," *ACM Transactions on Graphics*, vol. 30, no. 4, p. 69, 2011.

[41] H. G. Barrow, J. M. Tenenbaum, R. C. Bolles, and H. C. Wolf, "Parametric correspondence and chamfer matching: Two new techniques for image matching," in *Proc. 5th Int'l Joint Conf. Artificial Intelligence*, 1977, pp. 659–663.

[42] R. von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "LSD: A fast line segment detector with a false detection control," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 4, pp. 722–732, April 2010.

[43] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 679–698, 1986.

[44] [Online]. Available: https://sites.google.com/site/icdar15smartdoc/

초록

본 논문에서는 물체에 대한 모양 정보 학습을 이용하여 물체의 전체 혹은 일부 윤곽선을 추정하는 새로운 윤곽선 추적 방법을 베이지안(Bayesian) 기법에 기반하여 제안한다. 모양은 물체를 나타내는 매우 중요한 특징이기 때문에 이 정보를 활용하면 물체 추적 성능을 상당히 높일 수 있다. 제안하는 방법은 물체 윤곽선을 표현하기 위해 새로운 상태 공간 모델(state space model)을 만드는데, 이 상태 공간에서는 물체 윤곽선의 정형적 변형(rigid motion)과 비정형적 변형(non-rigid motion)을 독립적으로 나타낸다. 윤곽선의 정형적 변형은 수학적으로 잘 정의되어 있기 때문에 모양 정보 학습은 비정형적 변형에만 초점을 맞추면 되고, 이렇게 학습된 모델을 사용하면 여러 가지 복합적 모양 변형을 모두 표현할 수 있다. 게다가 안정적인 윤곽선 추적을 위해 물체의 경계 부분에 대한 대비, 물체의 모습, 시간적 연속성 등을 다각적으로 고려한 평가 함수(measurement function)를 제안한다. 개방, 폐쇄, 다중 윤곽선 등 여러 상황에 대한 윤곽선 추적 실험을 제안하는 방법을 이용하여 진행하였고, 각 상황에 따라 상태 공간 모델과 평가 함수는 조금씩 다르게 설계되었다.

첫째로, 개방 윤곽선 추적에 대해 실험하였는데, 개방 윤곽선 추적은 폐쇄 윤곽선이나 박스(box) 추적에 비해 상대적으로 적게 연구된 분야이다. 제안하는 상태 공간 모델로 개방 윤곽선을 쉽게 표현할 수 있고, 앞서 언급했듯이 윤곽선의 정형적 변형과 비정형적 변형은 완전히 분리되어 독립적으로 움직인다. 평가 함수는 대비(contrast), 지역적 추적(local track), 물체 모습(appearance)을 반영하도록 설계하여 안정적으로 윤곽선 추적이 되게 하였다. 사람의 오메가 형상(omega shape)과 치타의 옆머리 모습(cheetah profile)을 추적 대상으로 선정하여 이를 개방 윤곽선으로 나타내고 추적하는 실험을 진행하였으며, 과거에 연구된 윤곽선 추적 방법들에 비해 제안하는 방법이 나은 성능을 보임을 확인하였다. 또한 최근의 박스 추적 방법들과도 비교한 결과 제안하는 방법이 빠른 움직임 등 어려운 상황에서 윤곽선의 위치 추적에 더 강인하게 동작하였다.

둘째로, 제안하는 방법을 이용하여 폐쇄 윤곽선 추적을 하였는데, 이는 주로 물체 추출(segmentation)이나 레벨 셋 방법(level set method)을 이용하는 방식으로 연구되었다. 제안하는 상태 공간 모델을 이용하여 폐쇄 윤곽선을 나타내고, 이 윤곽선은 역학 모델(dynamic model)에 의하여 모양과 위치가 변형된다. 앞서 설계된 평가 함수 중 지역 추적 모델을 새롭게 다음과 같이 대체하는데, 물체의 부분 모습 모델을 지역 패치(local patch)와 윤곽선 중심에 대한 상대적 위치를 이용하여 만들고 이를 통해 지역 추적 모델을 계산한다. 자동차 블랙 박스에 찍힌 자동차들의 모습을 추적 대상으로 선정하여 폐쇄 윤곽선 추적을 실험 하였는데, 제안하는 방법이 기존 폐쇄 윤곽선 추적 방법들과 박스 추적 방법들에 비해 높은 성능을 보였다. 또한, 폐쇄 윤곽선의 다른 대상으로 문서를 선택하여 문서 포착 알고리즘도 새롭게 제안하였다. 문서의 모양은 사각형으로 잘 알려져 있기 때문에 이를 이용하여 문서의 경계를 찾아내는 방법을 제안하였고 과정은 다음과 같다. 우선 선분 검출기(line segment detector)를 이용하여 문서가 될 수 있는 여러 후보들을 만들고 이 후보들 중 가장 높은 평가 함수 값을 갖는 후보를 최종적으로 선택한다. 제안하는 방법을 2015 ICDAR competition에 제출하였고, 문서 포착에 대해 높은 평가를 받았다.

마지막으로, 다중 윤곽선 추적을 제안하는 방법으로 실험하였다. 다중 윤곽선은 서로 모습과 색깔이 비슷하여 같은 범주(category)에 속하는 물체들의 윤곽선들로 선택하였다. 같은 범주에 속하는 물체들은 서로 비슷하게 보이기 때문에 동시에 추적하는데 어려움이 있다. 물체들이 같은 범주에 속하기 때문에 학습된 하나의 모양 정보를 이용하여 제안하는 방법을 실험하였고, 다중 윤곽선이 하나의 상태 공간 모델로 표현되도록 상태 공간 모델을 수정하였다. 그리고 다중 윤곽선 추적을 잘 하기 위하여 물체들 사이의 상호 작용(interaction)에 관련된 모델을 만들고 이를 역학 모델에 덧붙여서 고려하였다. 사람의 두 다리를 추적할 대상으로 선택하여 이를 제안하는 방법과 기존의 다중 추적 방법들로 실험 하였는데, 기존 방법들은 사람이 걸으면서 두 다리가 겹칠 때 이를 잘 구분하지 못하였다. 그에 반해, 제안하는 방법은 새롭게 고려된 상호 작용 모델을 이용하여 이를 잘 구분해 내었다.

주요어 : 윤곽선 추적, 모양 정보, 개방 윤곽선, 다중 윤곽선

학 번 : 2010-20915